NASA CONTRACTOR REPORT

NASA CR-1722

CALCULATIVE TECHNIQUES FOR TRANSONIC FLOWS ABOUT CERTAIN CLASSES OF AIRFOILS AND SLENDER BODIES

by J. R. Spreiter and S. S. Stahara

Prepared by
NIELSON ENGINEERING AND RESEARCH, INC.
Mountain View, Calif.
for Ames Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1971

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-1722		

4. Title and Subtitle	5. Report Date
CALCULATIVE TECHNIQUES FOR TRANSONIC FLOWS ABOUT CERTAIN CLASSES OF AIRFOILS AND SLENDER BODIES	April 1971
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
J. R. Spreiter and S. S. Stahara	
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
Nielsen Engineering and Research, Inc. Mountain View, California	NAS 2-5410
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Contractor Report
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Procedures based on the method of local linearization and transonic equivalence rule were developed for predicting properties of transonic flows about thin airfoils and slender bodies. The procedures are applicable to transonic flows with free stream Mach numbers near one, below the lower critical and above the upper critical. Comparisons with experimental surface pressure distributions exhibit good agreement for all shapes considered, although discrepancies appear near the aft region, particularly for the slender bodies. Analysis of the discrepancies suggests substantial wind tunnel wall interference effects are present in the experimental results for the three-dimensional shapes. Effects of shock-wave boundary-layer interaction and vortex-induced separation are not included in the theory.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Transonic Flow Aerodynamics Airfoils Slender Bodies Transonic equivalence Rule Local Linerization	Unclassified-Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	145	$3.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

TABLE OF CONTENTS

CALCULATIVE TECHNIQUES FOR TRANSONIC FLOWS ABOUT CERTAIN CLASSES OF AIRFOILS AND SLENDER BODIES

By John R. Spreiter* and Stephen S. Stahara
Nielsen Engineering & Research, Inc.

SUMMARY

Analysis and the development of associated computational programs were carried out for the purpose of developing calculative techniques for predicting properties of transonic flows about thin airfoils and slender bodies. The procedures are based on the method of local linearization and the transonic equivalence rule and apply to transonic flows with free-stream Mach number in the ranges close to one, below the lower critical, and above the upper critical. Comparisons were made with experimental results for surface pressure distributions on a class of two-dimensional airfoils at $M_\infty = 1$ and for surface and flow-field pressure distributions for a number of slender, pointed, axisymmetric and nonaxisymmetric bodies, both nonlifting and lifting, at $M_\infty = 1$ and also at Mach numbers below the lower critical and above the upper critical.

Altogether, the calculated results exhibit good agreement with the data for all the shapes considered and over the major portion of the body length, although notable discrepancies consistently appear near the rear, particularly for the three-dimensional bodies. In addition to effects of shock-wave boundary-layer interaction and vortex-induced separation not included in the theory, analysis of the discrepancies suggests that substantial wind-tunnel wall interference effects are present in the experimental results for the three-dimensional shapes.

INTRODUCTION

Because of both the inherent difficulty of solving the governing equations of transonic aerodynamics and the low interest in problems characteristic to transonic flight, little substantial progress has been made in this important field for nearly a decade (refs. 1 and 2). Even

---

*Professor, Departments of Applied Mechanics and Aeronautics and Astronautics, Stanford University, Stanford, California. (Consultant at Nielsen Engineering & Research, Inc.).

with the recent attention given to transonic problems, it appears that the majority of research is currently directed at the two-dimensional case (for example, refs. 3, 4, and 5). In comparison, little emphasis has been placed on the analysis of transonic flow about three-dimensional bodies, particularly lifting, nonaxisymmetric shapes, even though all of the essential elements, both theoretical (refs. 6 and 7) and experimental (ref. 8) have been available for doing this for over a decade. In fact, prior to this investigation, there existed only one isolated application and experimental evaluation of the theory to a nonaxisymmetric, lifting flow at $M_\infty = 1$ (ref. 9). For axisymmetric flows, systematic studies of theoretical and experimental results have been done only for surface pressure distributions on cone-cylinders and parabolic-arc bodies (ref. 7), and for a pair of related bodies having maximum thickness at 30 percent and 70 percent of the body length (ref. 10). The present study extends the previous work to include nonaxisymmetric bodies, lifting, as well as nonlifting cases, and conditions in the flow field in addition to the body surface. Comparisons of the theoretical results with experimental data are made throughout.

Although the ultimate goal of the present investigation is to develop calculative techniques for the prediction of the flow field, pressure distribution, and aerodynamic characteristics of three-dimensional, lifting, wing-body combinations, the purposes of this initial study are (1) to select an appropriate theoretical method in view of the goal, and (2) to proceed to develop computational programs for calculating the pressure distributions on the surface and in the flow fields of increasingly complex classes of shapes. Because experimental verification of the theory is essential, primary attention is directed toward shapes for which data are available. Systematic comparisons of experimental and theoretical results provide both a thorough evaluation of the effectiveness of the theoretical method and also establish a basis for attacking the more general problem of predicting transonic flow about wing-body combinations.

LIST OF SYMBOLS

a	major axis of elliptic cross section of parabolic-arc bodies
a_1	local radius of body of wing-body combination
A	aspect ratio
b	minor axis of elliptic cross section of parabolic-arc bodies
c	wing chord
C	Euler's constant
C_p	pressure coefficient, $(p - p_\infty)/\frac{1}{2}\rho_\infty U_\infty^2$
$\overline{C}_p$	similarity form of pressure coefficient, eq. (3)
$\overline{C}_{p_a}$	similarity pressure coefficient immediately ahead of trailing-edge shock on two-dimensional airfoil
$\overline{C}_{p_a}^*$	limiting value for $\overline{C}_{p_a}$ for which flow is exactly sonic immediately downstream of a shock wave attached to the trailing edge of a two-dimensional airfoil, eq. (100)
$(\overline{C}_{p_a})_{att}$	limiting value for $\overline{C}_{p_a}$ for which the oblique shock at the trailing edge of a two-dimensional airfoil is able to turn the flow through the required angle for the shock to remain attached, eq. (99)
$\overline{C}_{p_b}$	similarity pressure coefficient immediately behind trailing-edge shock on two-dimensional airfoil
$(\overline{C}_{p_b})_{strong}$	value of $\overline{C}_{p_b}$ for which flow behind trailing-edge shock on two-dimensional airfoil is subsonic
$(\overline{C}_{p_b})_{weak}$	value of $\overline{C}_{p_b}$ for which flow behind trailing-edge shock on two-dimensional airfoil is supersonic
$(C_p)_B$	pressure coefficient associated with the equivalent body of a thin, elliptic, cone-cylinder wing, eq. (104)
C_{p_i}	incompressible pressure coefficient, eq. (5)
$(C_p)_w$	pressure coefficient associated with a thin, elliptic, cone-cylinder wing, eq. (104)

$(C_p)_{w,\ell}$	pressure coefficient associated with the lower surface of a thin, elliptic, cone-cylinder wing
$(C_p)_{w,u}$	pressure coefficient associated with the upper surface of a thin, elliptic, cone-cylinder wing
D	maximum diameter of a body of revolution
f_E	quantity defined by eq. (80)
f_H	quantity defined by eq. (85)
f_1	quantity associated with calculation of axisymmetric flow field, eq. (90)
g	quantity defined by eq. (58)
g_1	quantity associated with calculation of surface pressures on nonlifting, nonaxisymmetric, parabolic-arc bodies, eq. (91)
g_2	quantity associated with calculation of surface pressures on lifting, nonaxisymmetric, parabolic-arc bodies, eq. (92)
g_3	quantity associated with calculation of flow-field pressures for nonlifting, nonaxisymmetric, parabolic-arc bodies, eq. (93)
G	quantity defined by eq. (72)
H	quantity defined by eq. (73)
$\hat{i}, \hat{j}, \hat{k}$	unit vectors parallel to the x, y, z axes
k	equal to $M_\infty^2(\gamma + 1)/U_\infty$
ℓ	body length
m	tangent of semiapex angle of wing planform
M_∞	free-stream Mach number
$M_{cr,\ell}$	lower critical Mach number
$M_{cr,u}$	upper critical Mach number
n	exponent describing airfoil or body ordinates and related to location of point of maximum thickness, eqs. (7), (9), (77), (79)
n_1, n_2, n_3	direction cosines with respect to x, y, z axes

p_∞	free-stream pressure
Δp	aerodynamic loading
q_∞	free-stream dynamic pressure
r	radial distance in crossflow plane, $\sqrt{y^2 + z^2}$
R	radius of body of revolution
R_{eb}	radius of equivalent body of revolution
s	semispan of wing
S	area distribution of body of revolution
t	maximum thickness of thin cone-cylinder
u, v, w	perturbation velocity components parallel to the x, y, z axes, respectively
u_a, w_a	perturbation velocity components immediately upstream of trailing-edge shock on two-dimensional airfoil
u_b, w_b	perturbation velocity components immediately downstream of trailing-edge shock on two-dimensional airfoil
u_B, v_B, w_B	perturbation velocity components associated with solution for transonic flow about equivalent body of revolution
$u_{2,B}, v_{2,B}, w_{2,B}$	perturbation velocity components associated with two-dimensional incompressible solution of expansion or contraction of equivalent cross section in crossflow plane
$u_{2,t}, v_{2,t}, w_{2,t}$	perturbation velocity components associated with two-dimensional incompressible solution of expanding or contracting cross section in crossflow plane
$u_{2,\alpha}, v_{2,\alpha}, w_{2,\alpha}$	perturbation velocity components associated with two-dimensional incompressible solution of translating cross section in crossflow plane
U_∞	free-stream velocity
$\underline{V}$	total velocity vector
V_E	quantity defined by equation (81)
V_H	quantity defined by equation (86)

$W_{2,t}$	complex potential describing two-dimensional incompressible flow about expanding or contracting cross section in crossflow plane
$W_{2,\alpha}$	complex potential describing two-dimensional incompressible flow about translating cross section in crossflow plane
x, y, z	body-fixed Cartesian coordinate system with x axis directed rearward and aligned with longitudinal axis of body, y axis directed to the right facing forward, and z axis directed vertically upward
x_m	location of minimum of incompressible pressure coefficient, C_{p_i}
x_s	location of point closest to origin where $S''(x) = 0$
x*	location of sonic point on two-dimensional airfoil, eq. (12)
x_1	dummy variable
α	angle of attack
γ	ratio of specific heats
θ	polar angle in crossflow plane
λ	ratio of major to minor axes of elliptic cross section, a/b
λ_E	quantity defined by equation (84)
λ_H	quantity defined by equation (89)
Λ_E	quantity defined by equation (70)
Λ_H	quantity defined by equation (71)
ξ	dummy variable
ξ_∞	similarity parameter for two-dimensional transonic flow, $(M_\infty^2 - 1)/\left[M_\infty^2(\gamma + 1)\tau\right]^{2/3}$
ρ_∞	free-stream density
σ	complex variable in crossflow plane, y + iz
τ	thickness ratio

ϕ	perturbation velocity potential
ϕ_B	perturbation velocity potential associated with transonic flow about equivalent body of revolution
ϕ_2	perturbation velocity potential associated with two-dimensional incompressible solutions to translation and growth of cross section in crossflow plane
$\phi_{2,B}$	perturbation velocity potential associated with two-dimensional incompressible solution for expansion or contraction of equivalent cross section in crossflow plane
$\phi_{2,t}$	perturbation velocity potential associated with two-dimensional incompressible solution for expansion or contraction of cross section in crossflow plane
$\phi_{2,\alpha}$	perturbation velocity potential associated with two-dimensional incompressible solution for translation of cross section in crossflow plane
Φ	total velocity potential
ω	cone semiapex angle

THEORETICAL BASIS OF ANALYSIS

Methods Available

Of the methods currently available for obtaining solutions of transonic flows (ref. 1) only two were considered appropriate in view of the ultimate goal of this study. These are the integral equation method (refs. 11 through 18) and the method of local linearization (refs. 7, 19, and 20). Both of these methods are notable for their success in providing approximate solutions of good quality of the small disturbance equation for inviscid transonic flow,

$$(1 - M_\infty^2)\phi_{xx} + \phi_{yy} + \phi_{zz} = \frac{M_\infty^2(\gamma + 1)}{U_\infty}\phi_x\phi_{xx} \tag{1}$$

where M_∞ is the free-stream Mach number, U_∞ the free-stream velocity, γ the ratio of specific heats, ϕ the perturbation velocity potential, and the coordinate system is such that x is either in the direction of the free stream or, in the case of lifting flows, aligned with the longitudinal axis of the body. This equation differs from the well-known

Prandtl-Glauert equation of linearized theory only by the retention of the term on the right-hand side; this term, however, has the effect of making the equation both nonlinear and of mixed type. It has been amply demonstrated (refs. 21 through 24) that a significant body of transonic flow problems can be adequately treated within the framework of the theory represented by this equation. An important additional feature of equation (1) is that it is applicable not only to transonic flows, but to subsonic and supersonic flows as well. Consequently, it provides a basis for a unified flow theory for Mach numbers from 0 up to that Mach number (at least 2 or 3) at which nonlinear hypersonic effects become significant and must be taken into account.

With regard to the integral equation method, its theoretical basis stems from a nonlinear integral equation derived from equation (1) by application of Green's theorem. The method is such that for a body of specified shape the analysis proceeds directly to the solution (as opposed to the hodograph method) from initial analytical steps through a combination of numerical and iterative procedures. Although the method is potentially more versatile, it has been applied primarily to planar flow past nonlifting symmetric airfoils with sharp leading edges and for free-stream Mach numbers less than unity. Development of this method to handle a lifting, wing-body combination would require three difficult steps: (a) extension of the two-dimensional results for airfoils to wings of finite aspect ratio, (b) introduction of angle of attack, and (c) introduction of the body of the wing-body combination.

On the other hand, the local linearization method which has grown out of the parabolic method of Oswatitsch and Keune (ref. 25) has been developed for planar flow past thin, nonlifting, symmetric airfoils (ref. 19) and axisymmetric flow past slender bodies of revolution (ref. 7) for free-stream Mach numbers in the ranges $M_\infty \approx 1$, $M_\infty \leq M_{cr,\ell}$, and $M_{cr,u} \leq M_\infty$ (where $M_{cr,\ell}$ and $M_{cr,u}$ are the lower and upper critical Mach numbers bounding the transonic range). It has also been extended to flows with $M_\infty \approx 1$ past nonlifting wings of finite span having simple planform and airfoil shapes (ref. 20). Thus, it is the most versatile of all theoretical methods currently available for predicting the aerodynamic properties of thin wings and slender bodies in the transonic regime. In addition, this method has consistently demonstrated accuracy comparable with the best theoretical and experimental results available.

An important additional consideration is the possibility of further extending this method by generalizing the analysis to include shock waves. It appears possible that the local linearization method may be advanced in this direction as well as others without becoming analytically cumbersome. Accordingly, this method has been selected for further study as holding the greatest promise for successful application to three-dimensional transonic flows about lifting wing-body combinations.

Two-Dimensional Flows

The application of the local linearization method to two-dimensional flow past thin, nonlifting airfoils is carried out in detail in reference 19. Because for planar transonic flows, the similarity rules for relating flows about families of affinely related profiles are exact within the framework of transonic small-disturbance theory, it is both concise and convenient to present results for these classes of flows in terms of the transonic similarity parameter

$$\xi_\infty \equiv \frac{M_\infty^2 - 1}{\left[M_\infty^2(\gamma + 1)\tau\right]^{2/3}} \tag{2}$$

and the similarity form of the pressure coefficient

$$\overline{C}_p \equiv \left[\frac{M_\infty^2(\gamma + 1)}{\tau^2}\right]^{1/3} C_p \tag{3}$$

where τ represents the thickness ratio.

Once results are calculated in terms of these reduced variables, it is a simple matter to convert them for a specific airfoil to results in terms of physical variables and parameters by means of equation (3) where $C_p = -2\phi_x/U_\infty$.

Surface pressure distributions.- For nonlifting, purely subsonic flows, the local linearization method provides the result that the surface pressure distribution is given by

$$\overline{C}_p = 2\left\{\xi_\infty + \left[(-\xi_\infty)^{3/2} + \frac{3}{4}\,\frac{C_{p_i}}{\tau}\right]^{2/3}\right\} \tag{4}$$

where C_{p_i} is the well-known incompressible pressure coefficient

$$\frac{C_{p_i}}{\tau} = -\frac{2}{\pi} \int_0^c \frac{\frac{d(Z/\tau)}{d\xi}}{x - \xi} d\xi \tag{5}$$

$\int$ signifies the Cauchy principal, and $dZ/d\xi$ represents the slope of the airfoil upper surface. The two-dimensional shapes considered herein are those tested in reference 27. These profiles are members of the family of airfoils having ordinates Z given by

$$\frac{Z}{c} = \frac{\tau n^{n/(n-1)}}{2(n-1)} \left[\frac{x}{c} - \left(\frac{x}{c} \right)^n \right] \tag{6}$$

$$\left(\frac{x}{c} \right)_{Z_{max}} = \left(\frac{1}{n} \right)^{1/(n-1)} \tag{7}$$

or

$$\frac{Z}{c} = \frac{\tau n^{n/(n-1)}}{2(n-1)} \left[1 - \frac{x}{c} - \left(1 - \frac{x}{c} \right)^n \right] \tag{8}$$

$$\left(\frac{x}{c} \right)_{Z_{max}} = 1 - \left(\frac{1}{n} \right)^{1/(n-1)} \tag{9}$$

for n = constant ≥ 2.

For purely supersonic flows, the surface pressure distribution according to local linearization theory is

$$\overline{C}_p = 2 \left\{ \xi_\infty - \left[\xi_\infty^{3/2} - \frac{3}{2} \frac{d}{dx} \left(\frac{Z}{\tau} \right) \right]^{2/3} \right\} \tag{10}$$

while the corresponding result for accelerating transonic flows with $M_\infty \approx 1$ is

$$\overline{C}_p = 2 \left(\xi_\infty - \left\{ \frac{3}{\pi} \int_{x^*}^{x} \left[\frac{d}{dx_1} \int_0^{x_1} \frac{d(Z/\tau)/d\xi}{\sqrt{x_1 - \xi}} d\xi \right]^2 dx_1 \right\}^{1/3} \right) \tag{11}$$

where x^* is the point at which

$$\frac{d}{dx} \int_0^x \frac{d(Z/\tau)/d\xi}{\sqrt{x - \xi}} \, d\xi = 0 \tag{12}$$

Critical Mach numbers.- The lower and upper critical Mach numbers are defined, respectively, as the lowest subsonic Mach number and the highest supersonic Mach number at which sonic velocity occurs on the surface of the body. For a given airfoil in subsonic flow, as the free-stream Mach number is increased, sonic velocity first occurs on the body at the point of minimum pressure and at that free-stream Mach number at which

$$\overline{C}_p = \overline{C}_p\Big|_{cr,\ell} = 2\xi_\infty \tag{13}$$

Thus, at $M_\infty = M_{cr,\ell}$ equation (4) provides

$$(-\xi_\infty)^{3/2} + \frac{3}{4} \left.\frac{C_{p_i}}{\tau}\right|_{x=x_m} = 0 \tag{14}$$

where x_m is the point at which

$$\frac{C_{p_i}}{\tau} = \left(\frac{C_{p_i}}{\tau}\right)_{min} \tag{15}$$

This yields the following equation to be solved for $M_{cr,\ell}$:

$$(1 - M_{cr,\ell}^2)^{3/2} + \frac{3}{4} (\gamma + 1)\tau \left(\frac{C_{p_i}}{\tau}\right)_{min} M_{cr,\ell}^2 = 0 \tag{16}$$

For a given sharp-nosed airfoil, as the free-stream Mach number M_∞ increases beyond one, the bow shock wave that is formed ahead of the airfoil nose moves steadily toward attachment to it and the region of subsonic flow surrounding the nose continuously diminishes in size. The upper critical Mach number is attained when the region of subsonic flow first disappears. According to the method of local linearization, this occurs when the quantity Λ_H vanishes, where

$$\Lambda_H = \xi_\infty^{3/2} \mp \frac{3}{2} \frac{d}{dx}\left[\frac{Z(x)}{\tau}\right] \tag{17}$$

and the upper and lower signs are to be applied to the upper and lower surfaces, respectively. Thus, for the upper surface of the airfoil, the location at which Λ_H would first vanish is at the point of largest positive slope, whereas on the lower surface it would be at the point of largest negative slope. For symmetric airfoils, these points coincide, and in particular, for the airfoils considered herein, this location is at the nose. From equation (10), this implies that

$$\xi_\infty\Big|_{cr,u}^{3/2} - \frac{3}{2} \frac{d}{dx}\left(\frac{Z}{\tau}\right)\Big|_{x=o} = 0 \tag{18}$$

which provides the following equation for $M_{cr,u}$:

$$(M_{cr,u}^2 - 1)^{3/2} - \frac{3}{2} (\gamma + 1)\tau \left\{\frac{d}{dx}\left(\frac{Z}{\tau}\right)\Big|_{x=o}\right\} M_{cr,u}^2 = 0 \tag{19}$$

Three-Dimensional Flows

The analysis presented for the three-dimensional flows considered is expressed in terms of a body-fixed Cartesian coordinate system centered at the nose with the x axis directed rearward and aligned with the longitudinal axis of the body, the y axis directed to the right facing forward, and the z axis directed vertically upward so that the x-z plane is a plane of symmetry of the body - for example, perpendicular to the plane of the wing of a wing-body combination. The free-stream direction may be inclined a small angle to the x axis, although attention is confined to cases in which it is in the x-z plane, i.e., no sideslip. With the fundamental assumption of inviscid small disturbance flow theory that the flow is irrotational and isentropic so that the velocity $\underline{V}$ at any point can be obtained as the gradient of a potential Φ, it is possible to define a perturbation velocity potential ϕ related to Φ according to (see ref. 6)

$$\Phi(x,y,z) = U_\infty(x + \alpha z) + \phi(x,y,z) \tag{20}$$

where U_∞ represents the free-stream velocity, and α the angle of attack. Although the governing partial differential equation for ϕ has

been written in several slightly different forms in transonic flow studies, we use the form given in equation (1) for the reasons first put forward in reference 27 and confirmed in subsequent studies. As indicated in reference 6, this equation applies whether the coordinate system is aligned with the x axis parallel to the direction of the free stream, as in most derivations (ref. 28), or inclined a small angle to it as in the present applications to lifting configurations. The expression for the pressure coefficient $C_p = (p - p_\infty)/(\rho_\infty U_\infty^2/2)$ is not invariant with respect to small rotations of the coordinate system, however, and is as follows (ref. 6) in the coordinate system described above

$$C_p = -\frac{2}{U_\infty}(\phi_x + \alpha\phi_z) - \frac{1}{U_\infty^2}(\phi_y^2 + \phi_z^2) \tag{21}$$

The boundary conditions require that $\underset{\sim}{V} = i\,U_\infty + k\,\alpha U_\infty$ infinitely far from the body, where $\hat{i}$, $\hat{j}$, and $\hat{k}$ are unit vectors parallel to the x, y, and z axes, and that the velocity component V_n normal to the body surface be zero at the body. The boundary conditions for ϕ may thus be written as follows for slender bodies or thin wings having small n_1 (ref. 6):

$$\left.\begin{aligned} \phi(\infty) &= 0 \\ U_\infty(n_1 + \alpha n_3) + n_2\phi_y + n_3\phi_z &= U_\infty(n_1 + \alpha n_3) + \phi_n \\ &= 0 \quad \text{(on the body)} \end{aligned}\right\} \tag{22}$$

where $n = \hat{i}\,n_1 + \hat{j}\,n_2 + \hat{k}\,n_3$ is the unit normal to the surface, and n_1, n_2, and n_3 are the direction cosines of n with respect to the x, y, and z axes, respectively.

Axisymmetric Flows

Application of the local linearization method to axisymmetric flows about slender bodies of revolution has been carried out and is given in detail in reference 7. Because the similarity rules for relating surface pressures on families of affine bodies of revolution are not exact within the framework of the small disturbance theory for transonic flow, in contrast to the situation for airfoils and wings of finite span, it is convenient in this instance to calculate flow properties in terms of

the ordinary pressure coefficient C_p and the perturbation velocity component u.

Surface pressure distributions.- The local linearization method provides the following results for the perturbation velocity component u/U_∞ on the body surface. For purely subsonic flows

$$\frac{d}{dx}\left(\frac{u}{U_\infty}\right) = \frac{S'''(x)}{4\pi} \ln\left(1 - M_\infty^2 - ku\right) + \frac{d}{dx}\left[\frac{S''(x)}{4\pi} \ln \frac{S(x)}{4\pi x(\ell - x)} + \frac{1}{4\pi} \int_0^\ell \frac{S''(x) - S''(\xi)}{|x - \xi|}\, d\xi\right] \tag{23}$$

for purely supersonic flows

$$\frac{d}{dx}\left(\frac{u}{U_\infty}\right) = \frac{S'''(x)}{4\pi} \ln\left(M_\infty^2 - 1 + ku\right) + \frac{d}{dx}\left[\frac{S''(x)}{4\pi} \ln \frac{S(x)}{4\pi x^2} + \frac{1}{2\pi} \int_0^x \frac{S''(x) - S''(\xi)}{x - \xi}\, d\zeta\right] \tag{24}$$

and for accelerating transonic flows with $M_\infty \approx 1$

$$\frac{d}{dx}\left(\frac{u}{U_\infty}\right) = \frac{S'(x)S''(x)}{4\pi S(x)} + \exp\left\{\frac{4\pi}{S''(x)}\left[\frac{u}{U_\infty} + \frac{M_\infty^2 - 1}{M_\infty^2(\gamma + 1)} - \frac{S''(x)}{4\pi} \ln \frac{M_\infty^2(\gamma + 1)S(x)e^C}{4\pi x} - \frac{1}{4\pi} \int_0^x \frac{S''(x) - S''(\xi)}{x - \xi}\, d\xi\right]\right\} \tag{25}$$

where k in equations (23) and (24) is equal to $M_\infty^2(\gamma + 1)/U_\infty$, C in equation (25) is Euler's constant ≈ 0.5772, $S(x)$ represents the area distribution, and primes indicate differentiation with respect to the appropriate variable.

For general bodies, closed form solutions analogous to equations (4), (10), and (11) for planar flow cannot be found and recourse must be had to numerical integration. However, equations (23), (24), and (25) are of the general form

$$\frac{du}{dx} = F(x,u) \tag{26}$$

so that standard numerical techniques can be applied. For all bodies to be considered herein, the integration is begun at the positive root, x_s, of the equation

$$S''(x) = 0 \tag{27}$$

that is closest to the origin.

The values of u/U_∞ at this point that are necessary to start the integration are for subsonic flow (eq. (23))

$$\frac{u}{U_\infty} = \frac{1}{4\pi} \int_0^\ell \frac{S''(x) - S''(\xi)}{|x - \xi|} d\xi \quad \text{at} \quad S''(x) = 0 \tag{28}$$

for supersonic flow (eq. (24))

$$\frac{u}{U_\infty} = \frac{1}{2\pi} \int_0^x \frac{S''(x) - S''(\xi)}{x - \xi} d\xi \quad \text{at} \quad S''(x) = 0 \tag{29}$$

and for accelerating transonic flows with $M_\infty \approx 1$ (eq. (25))

$$\frac{u}{U_\infty} = \frac{1 - M_\infty^2}{M_\infty^2(\gamma + 1)} + \frac{1}{4\pi} \int_0^x \frac{S''(x) - S''(\xi)}{x - \xi} d\xi \quad \text{at} \quad S''(x) = 0 \tag{30}$$

The boundary conditions (28) and (29) are sufficient to determine a unique solution for the cases of purely subsonic or purely supersonic flow, respectively, but additional considerations are necessary for accelerating transonic flows. This is so because the differential equation (25) is singular at the point x_s where $S''(x)$ vanishes. Consequently, there exist an infinite number of integral curves which pass through that point satisfying the boundary condition (30). Of all these curves, however, only one is analytic (all derivitives finite) and selection of it suffices to determine a unique solution that is in good agreement with experimental data. This choice assures that the solution for u/U_∞ can be expanded in a Taylor series in the neighborhood of the

point where $S''(x)$ vanishes. The remainder of the solution can then be determined by application of standard numerical techniques.

Critical Mach numbers.- The lower critical Mach number for slender axisymmetric bodies is found in a manner somewhat similar to that used for two-dimensional flows over thin airfoils. For a given body of revolution, the lower critical Mach number is the lowest subsonic free-stream Mach number at which sonic velocity occurs at the point on the body where $u/U_\infty = (u/U_\infty)_{max}$. This occurs at that free-stream Mach number which satisfies the relation

$$1 - M_{cr,\ell}^2 - M_{cr,\ell}^2(\gamma + 1)\left(\frac{u}{U_\infty}\right)_{max} = 0 \tag{31}$$

It is not possible to give a more explicit formula, however, because $(u/U_\infty)_{max}$ and M_∞ are related to each other through the differential equation (23) rather than an integral relationship such as (14) in the two-dimensional case. Thus, it is necessary to carry out a combination of numerical integrations of that equation and iterations on the free-stream Mach number in order to determine $M_{cr,\ell}$. Correspondingly, the upper critical Mach number is obtained at the highest supersonic free-stream Mach number such that for the class of shapes considered herein sonic velocity is obtained at the nose. At this point, $M_{cr,u} = M_\infty$ and

$$M_{cr,u}^2 - 1 + M_{cr,u}^2(\gamma + 1)\left(\frac{u}{U_\infty}\right)\Bigg|_{x=0} = 0 \tag{32}$$

Again, a more explicit relation cannot be given, as M_∞ and $(u/U_\infty)\big|_{x=0}$ are related through the differential equation (24) and a numerical integration of that equation together with an iteration procedure on M_∞ is necessary in order to obtain $M_{cr,u}$.

Transonic Equivalence Rule

The transonic equivalence rule provides a powerful tool for reducing a complex transonic flow to a sum of simpler constituent flows that are more amenable to analysis. The rule relates the transonic flow around a slender body of arbitrary cross section to the flow around an "equivalent" nonlifting body of revolution having the same longitudinal distribution

of cross sectional area $S(x)$. First stated by Oswatitsch (ref. 29) for transonic flow past thin nonlifting wings, it was extended to lifting wings by Spreiter (ref. 21) and to slender wing-body combinations of arbitrary cross section by Heaslet and Spreiter (ref. 6). This rule is obviously closely associated with the transonic area rule of Whitcomb (ref. 30) relating to drag, but pertains in addition to the properties of the flow field, such as the velocity and pressure, that are derivable from a knowledge of the potential. It is also closely related to one of the simplest results of slender-body theory of subsonic and supersonic, as well as transonic, flow, which states that the expression for ϕ in the vicinity of a slender body of arbitrary cross section is approximately of the form

$$\phi = \phi_2(x,y,z) + g(x) \tag{33}$$

where ϕ_2 is the solution of Laplace's equation

$$\phi_{yy} + \phi_{zz} = 0 \tag{34}$$

for the given boundary conditions in the y-z plane at each x station, and $g(x)$ is an additional contribution dependent upon M_∞ and $S(x)$ but not on the shape of the cross section. Consequently, it is possible to determine $g(x)$ from the simpler problem of axisymmetric flow past the equivalent body. Aside from the significant reductions of the transonic drag rise achieved by application of Whitcomb's area rule, the opportunities for advance provided by the transonic equivalence rule have never been fully exploited. This was primarily because methods were not available for solving the axisymmetric flow problems until several years after the equivalence rule was discovered; by then interest had declined to such a low level that few applications were made.

The equivalence rule is described mathematically by

$$\phi = \phi_{2,\alpha} + \phi_{2,t} - \phi_{2,B} + \phi_B \tag{35}$$

where $\phi_{2,\alpha}$, $\phi_{2,t}$, and $\phi_{2,B}$ are solutions of the two-dimensional Laplace equation as indicated in figure 1. Thus, $\phi_{2,\alpha}$ is the two-dimensional incompressible-flow solution for translation of the cross section and $\phi_{2,t}$ is the corresponding solution for growth of the cross section.

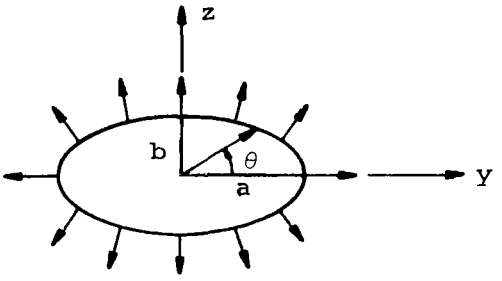
The term $\phi_{2,B}$ represents the two-dimensional incompressible-flow solution for the growth of a body of revolution having the same cross sectional area as the original body. The subtraction of this term in equation (35) has the effect of cancelling the logarithmic growth of $\phi_{2,t}$ at large lateral distances from the body axis. Finally, ϕ_B represents the three-dimensional solution of the full transonic equation (1). The above description for ϕ follows immediately from equation (33) by writing that equation once for the body of arbitrary cross section and once for the equivalent body, and then subtracting the results. The order of error in equation (35) has been established (ref. 2) for thin wings of aspect ratio A, chord c, and thickness ratio τ. It was found that the magnitude of the quantity $\phi/U_\infty c$ retained in the equivalence rule is $O(A\tau \ln A)$, whereas that of the quantities discarded in the derivation for $M_\infty = 1$ is $O(A^4\tau^2 \ln A)$. Since the magnitude of the quantities discarded in the derivation of the corresponding result in linearized subsonic and supersonic flow past slender bodies is $O(A^3\tau \ln A)$, it follows that the equivalence rule ought to be applicable to wings of greater aspect ratio at $M_\infty = 1$ than at any other Mach number.

Once the appropriate expression has been constructed for ϕ, the pressure distribution on or near the surface of slender bodies may be determined by use of the expression for C_p given in equation (21). The results may, in turn, be integrated to obtain expressions or values for the total forces, including lift, drag, and moments on slender bodies or wing-body combinations of arbitrary cross section. Since the aerodynamic loading, lift, and all lateral forces and moments may be expressed in terms of differences in pressure between pairs of points at the same longitudinal station, these quantities depend solely on ϕ_2 and are therefore independent of M_∞. In particular, we note, as discussed more than two decades ago (refs. 31, 32) and previous to the discovery of the transonic equivalence rule, that these quantities may be calculated quite adequately by linearized slender-body theory even though M_∞ may be unity.

One of the more important objectives of this investigation is to provide a systematic evaluation of the effectiveness of the transonic equivalence rule. Two basic tests exist for doing this. They involve study of the capability of the equivalence rule (a) to account for changes in flow properties on the surface of slender bodies due to

moderate nonaxisymmetric effects of body shape and lift and (b) to predict transonic flow-field properties at field points removed by moderate lateral distances from the body surface.

Surface pressure distributions - nonlifting bodies.- To study the effects of body shape on the flow properties at the surface of slender bodies, we have considered the nonlifting parabolic-arc bodies having elliptical cross sections studied experimentally in reference 8. The expression for the velocity potential $\phi_{2,t}$ required to describe the expanding and contracting of the elliptic cross sections shown in the figure below



while retaining a constant ratio $\lambda = a/b$ of major to minor axis is known (ref. 33) and given by the relation

$$\phi_{2,t} = \text{R.P.}(W_{2,t}) = \text{R.P.}\left\{\frac{U_\infty S'(x)}{2\pi} \ln\left[\frac{\sigma + (\sigma^2 - a^2 + b^2)^{1/2}}{2}\right]\right\} \tag{36}$$

where $W_{2,t}$ is the complex potential, R.P. signifies the real part of a complex function, σ is the complex variable in the crossflow plane

$$\sigma = y + iz \tag{37}$$

and $S'(x)$ is the first derivative of the cross sectional area

$$S(x) = \pi a(x)b(x) = \pi a^2(x)/\lambda \tag{38}$$

The velocity components associated with this flow are

$$u_{2,t} = \frac{\partial \phi_{2,t}}{\partial x} = \text{R.P.}\ \frac{\partial W_{2,t}}{\partial x} \tag{39}$$

$$(v - iw)_{2,t} = \frac{dW_{2,t}}{d\sigma} \tag{40}$$

Introduction of equation (36) into equations (39) and (40) provides the following results for these quantities at points on the body surface:

$$u_{2,t}\Big|_{surface} = \frac{U_\infty S''(x)}{4\pi} \ln\left[\frac{(\lambda + 1)^2 S(x)}{4\pi\lambda}\right] - U_\infty \left(\frac{dR_{eb}}{dx}\right)^2 (\lambda - 1)\left[\frac{\cos^2\theta - \lambda^3 \sin^2\theta}{\cos^2\theta + \lambda^4 \sin^2\theta}\right] \tag{41}$$

$$v_{2,t}\Big|_{surface} = \frac{U_\infty \sqrt{\lambda}\left(\frac{dR_{eb}}{dx}\right)(\cos^2\theta + \lambda^2 \sin^2\theta)^{1/2} \cos\theta}{(\cos^2\theta + \lambda^4 \sin^2\theta)} \tag{42}$$

$$w_{2,t}\Big|_{surface} = \frac{U_\infty \lambda^{5/2}\left(\frac{dR_{eb}}{dx}\right)(\cos^2\theta + \lambda^2 \sin^2\theta)^{1/2} \sin\theta}{(\cos^2\theta + \lambda^4 \sin^2\theta)} \tag{43}$$

where R_{eb} represents the radius of the equivalent body of revolution.

The corresponding expressions for $\phi_{2,B}$ are found by setting $\lambda = 1$ in the above formulas. Thus,

$$\phi_{2,B} = \frac{U_\infty S'(x)}{2\pi} \ln r \tag{44}$$

$$u_{2,B}\Big|_{surface} = \frac{U_\infty S''(x)}{4\pi} \ln\left[\frac{S(x)}{4\pi}\right] \tag{45}$$

$$v_{2,B} = U_\infty \left(\frac{dR_{eb}}{dx}\right) \cos\theta \tag{46}$$

$$w_{2,B} = U_\infty \left(\frac{dR_{eb}}{dx}\right) \sin\theta \tag{47}$$

The velocity perturbation $u_B = (\phi_B)_x$ associated with the three-dimensional solution for flow about the equivalent body of revolution is, of course, obtained through equation (23), (24), or (25), depending upon the Mach

number regime, while the lateral velocities are obtained from the slender-body approximation to the boundary condition for this problem, i.e.,

$$\lim_{r \to o} \left[r (\phi_B)_r \right] \simeq U_\infty R_{eb} \left(\frac{dR_{eb}}{dx} \right) \tag{48}$$

Thus, at the surface

$$(\phi_B)_r \Big|_{surface} = U_\infty \left(\frac{dR_{eb}}{dx} \right) \tag{49}$$

The pressure coefficient for points on the surface of this class of shapes can now be found by introducing the results of equations (41) through (49) into equation (21) with $\alpha = 0$. We find, after some simplification,

$$C_p \Big|_{surface} = - \frac{2u_B}{U_\infty} \Bigg|_{surface} - \frac{S''(x)}{2\pi} \ln \left[\frac{(\lambda + 1)^2}{4\lambda} \right]$$

$$- \left(\frac{dR_{eb}}{dx} \right)^2 \left[2 - \lambda \frac{\cos^2 \theta + \lambda^2 \sin^2 \theta}{\cos^2 \theta + \lambda^4 \sin^2 \theta} \right] \tag{50}$$

where u_B is found from the appropriate local linearization equation for axisymmetric flow about the equivalent parabolic-arc body of revolution.

Surface pressure distributions - lifting bodies.- The lifting transonic flows considered involve the same bodies mentioned above but now inclined to the free stream at small angles of attack. The two-dimensional potential $\phi_{2,\alpha}$ required to describe the falling, at a constant rate $U_\infty \alpha$, of the elliptical cross section is (see ref. 33)

$$\phi_{2,\alpha} = \text{R.P.}(W_{2,\alpha}) = \text{R.P.} \left\{ \frac{i\alpha U_\infty}{2} \left[\sigma - (\sigma^2 - a^2 + b^2)^{1/2} + \frac{(a + b)^2}{\sigma + (\sigma^2 - a^2 + b^2)^{1/2}} \right] \right\} \tag{51}$$

The velocity components at the body surface are found by operations analogous to those used for the thickness problem. Thus,

$$(u_{2,\alpha})\Big|_{\text{surface}} = \alpha U_\infty \frac{dR_{eb}}{dx} \frac{\lambda^{5/2}(\lambda + 1)(\cos^2\theta + \lambda^2 \sin^2\theta)^{1/2} \sin\theta}{(\cos^2\theta + \lambda^4 \sin^2\theta)} \tag{52}$$

$$v_{2,\alpha}\Big|_{\text{surface}} = \frac{-\alpha U_\infty \lambda^2(\lambda + 1) \sin\theta \cos\theta}{\cos^2\theta + \lambda^4 \sin^2\theta} \tag{53}$$

$$w_{2,\alpha}\Big|_{\text{surface}} = \frac{-\alpha U_\infty \lambda(\lambda^3 \sin^2\theta - \cos^2\theta)}{\cos^2\theta + \lambda^4 \sin^2\theta} \tag{54}$$

Upon using equation (21) for the pressure coefficient and equations (36), (44), and (51) for the various velocity potentials, the result for the surface pressure coefficient on lifting parabolic-arc bodies having elliptical cross sections is

$$\begin{aligned} C_p = -2 \frac{u_B}{U_\infty}\Big|_{\text{surface}} &- \frac{S''(x)}{2} \ln\left[\frac{(\lambda + 1)^2}{4\lambda}\right] \\ &- \left(\frac{dR_{eb}}{dx}\right)^2 \left\{2 - \lambda\left[\frac{\cos^2\theta + \lambda^2 \sin^2\theta}{\cos^2\theta + \lambda^4 \sin^2\theta}\right]\right\} \\ &- 2\alpha \left(\frac{dR_{eb}}{dx}\right) \frac{\lambda^{5/2}(\lambda + 1)(\cos^2\theta + \lambda^2 \sin^2\theta)^{1/2} \sin\theta}{(\cos^2\theta + \lambda^4 \sin^2\theta)} \\ &- \alpha^2 \left[\frac{-\lambda \sin^2\theta + (\lambda^2 + 2\lambda)\cos^2\theta}{\cos^2\theta + \lambda^4 \sin^2\theta}\right] \end{aligned} \tag{55}$$

For a body of revolution, this reduces to

$$C_p = -2 \frac{u_B}{U_\infty} - \left(\frac{dR_{eb}}{dx}\right)^2 - 4\alpha \left(\frac{dR_{eb}}{dx}\right) - \alpha^2(-1 + 4\cos^2\theta) \tag{56}$$

which agrees with reference 34.

Flow-field pressure distributions.- To provide a more complete test of the effectiveness of the equivalence rule for predicting transonic flow-field properties, flows about both axisymmetric and nonaxisymmetric shapes were considered. The axisymmetric bodies chosen for comparison of theory and data were those tested in references 35 and 36 having various locations of the point of maximum thickness, while the nonaxisymmetric shapes were the parabolic-arc bodies with elliptical cross sections mentioned previously in connection with surface pressure distributions. Unfortunately, flow-field data are available only for nonlifting flows about these bodies so that no systematic comparisons can be made for the important case of the transonic flow field about lifting bodies.

For an arbitrary nonlifting body, we may write according to equation (35)

$$\phi = \phi_{2,t} - \phi_{2,B} + \phi_B \tag{57}$$

where from equation (33)

$$g(x) = \lim_{r \to o} (\phi_B - \phi_{2,B}) \tag{58}$$

Equation (57) is valid both at points on and in the vicinity of the slender body. If the derivative of equation (57) is taken and the result applied to an arbitrary point in the flow field and also to a point on the body surface, we have

$$\phi_x\Big|_{field} = (\phi_{2,t})_x\Big|_{field} + (-\phi_{2,B} + \phi_B)_x\Big|_{field} \tag{59}$$

$$\phi_x\Big|_{surface} = (\phi_{2,t})_x\Big|_{surface} + (-\phi_{2,B} + \phi_B)_x\Big|_{surface} \tag{60}$$

Subtracting equations (59) and (60) and using equation (58), we have

$$\phi_x\Big|_{field} = \phi_x\Big|_{surface} + (\phi_{2,t})_x\Big|_{field} - (\phi_{2,t})_x\Big|_{surface} \tag{61}$$

and by using equation (59),

$$\phi_x\Big|_{field} = (\phi_{2,t})_x\Big|_{field} - (\phi_{2,B})_x\Big|_{surface} + (\phi_B)_x\Big|_{surface} \tag{62}$$

or equivalently,

$$\phi_x\Big|_{\text{field}} = u_{2,t}\Big|_{\text{field}} - \frac{U_\infty S''(x)}{2\pi} \ln R_{eb} + u_B\Big|_{\text{surface}} \tag{63}$$

Introduction of these expressions into equation (21) provides the result that the flow-field pressure coefficient for flow about a nonlifting, nonaxisymmetric slender body is

$$C_p = -\frac{2u_B}{U_\infty}\Big|_{\text{surface}} - \frac{2u_{2,t}}{U_\infty}\Big|_{\text{field}} + \frac{S''(x)}{\pi} \ln R_{eb} - \left[\frac{(\phi_{2,t})_y^2 + (\phi_{2,t})_z^2}{U_\infty^2}\right] \tag{64}$$

For a body of revolution

$$\phi_{2,t} = \frac{U_\infty S'(x)}{2\pi} \ln r \tag{65}$$

so that the pressure coefficient at a field point at a lateral distance r from the body axis is

$$C_p = -\frac{2u_B}{U_\infty}\Big|_{\text{surface}} - \frac{S''(x)}{\pi} \ln\left(\frac{r}{R_{eb}}\right) - \left(\frac{R_{eb}}{r}\right)^2 \left(\frac{dR_{eb}}{dx}\right)^2 \tag{66}$$

For the parabolic-arc bodies having elliptic cross section, $\phi_{2,t}$ is given in equation (36). Upon introducing that quantity in equation (64), we find that the pressure coefficient at an arbitrary point $re^{i\theta}$ in the crossflow plane is

$$C_p\Big|_{\sigma=re^{i\theta}} = -\frac{2u_B}{U_\infty}\Big|_{\text{surface}} + \text{R.P.}\left(\frac{-S''(x)}{\pi} \ln\left\{\frac{1}{2\sqrt{\lambda}}\left[\frac{r}{b} e^{i\theta} + \left(\frac{r^2}{b^2} e^{2i\theta} - \lambda^2 + 1\right)^{1/2}\right]\right\}\right)$$

(eq. (67) cont. on next page)

$$+\left(\frac{dR_{eb}}{dx}\right)^2\left\{\frac{2(\lambda^2-1)}{\left[\frac{r}{b}e^{i\theta}+\left(\frac{r^2}{b^2}e^{2i\theta}-\lambda^2+1\right)^{1/2}\right]\left[\frac{r^2}{b^2}e^{2i\theta}-\lambda^2+1\right]^{1/2}}\right.$$

$$\left.-\frac{\lambda}{\left[\frac{r^4}{b^4}+(\lambda^2-1)^2\right]^{1/2}}\right\}\Bigg) \tag{67}$$

COMPUTATIONAL ANALYSIS

This section furnishes a brief description of the various computational programs developed in order to provide an understanding of the basic logic as well as some of the limitations inherent in them. Listings of all programs are given in Appendices A and B.

Two-Dimensional Flows

All of the two-dimensional programs developed involve the class of airfoils whose ordinates and location of the point of maximum thickness are described by equations (6) through (9); hence, the first step in each of these programs involves the calculation of the exponent n according to either of equation (7) or (9).

Surface pressures.- The program developed for calculating the surface pressure distribution for purely subsonic flows then calculates the incompressible pressure coefficient C_{p_i} defined by equation (5). Because of the singularity at the point $x = \xi$, that particular form of the relation is inconvenient for numerical purposes. An alternative form more suited for numerical computation is found by adding to and subtracting from that equation the term

$$\frac{2}{\pi}\,\frac{dZ(x)/\tau}{dx}\oint_0^c\frac{d\xi}{x-\xi}=\frac{2}{\pi}\,\frac{dZ(x)/\tau}{dx}\ln\left(\frac{c-x}{x}\right) \tag{68}$$

Consequently, we have

$$\frac{C_{p_i}}{\tau}=-\frac{2}{\pi}\int_0^c\frac{\frac{dZ(\xi)/\tau}{d\xi}-\frac{dZ(x)/\tau}{dx}}{x-\xi}\,d\xi+\frac{2}{\pi}\,\frac{dZ(x)/\tau}{dx}\ln\left(\frac{c-x}{x}\right) \tag{69}$$

where the Cauchy principal value of the integral is no longer required, as the integrand is now continuous at all points in the integration interval. The integral involved in equation (69) is evaluated by using Simpson's rule. Thus, for a given airfoil of the class described by equations (6) and (8), the computational program determines C_{p_i}/τ from equation (69) for successive values of x along the airfoil surface and then calculates the values of $\overline{C}_p$ and C_p from equations (4) and (3), respectively. If the value of the quantity

$$\Lambda_E = (-\xi_\infty)^{3/2} + \frac{3}{4} \frac{C_{p_i}}{\tau} \tag{70}$$

becomes negative at some point along the airfoil surface, then the free-stream Mach number is greater than the lower critical Mach number; the flow field will possess a region of supersonic flow that may or may not be terminated by a shock wave and the above program does not apply.

The program for calculating the surface pressure distribution for purely supersonic flow involves calculation of the slope of the profile from either of equations (6) or (8) and the values of $\overline{C}_p$ and C_p from equations (10) and (3), respectively. If the quantity

$$\Lambda_H = \xi_\infty^{3/2} - \frac{3}{2} \frac{d}{dx} \left[\frac{Z(x)}{c} \right] \tag{71}$$

becomes negative at some point along the airfoil surface, the free-stream Mach number is less than the upper critical. Under these circumstances, a region of subsonic flow would then exist in the vicinity of the nose and the above program would not be applicable. Since the airfoils considered herein have their largest positive slope at the nose, the location at which Λ_H defined in equation (71) would first become negative would be at that point.

To calculate accelerating transonic flows at $M_\infty \approx 1$, it is convenient to recast equation (11) into an alternative form which avoids the square-root singularity of the integrand at $x_1 = \xi$. For this purpose, the inner integral of that equation can be rewritten in the following manner:

$$G(x_1) = \int_0^{x_1} \frac{\frac{dZ(\xi)/\tau}{d\xi}}{(x_1 - \xi)^{1/2}}\, d\xi$$

$$= \int_0^{x_1} \frac{\frac{dZ(\xi)/\tau}{d\xi} - \frac{dZ(x_1)/\tau}{dx_1}}{(x_1 - \xi)^{1/2}} + 2x_1^{1/2}\, \frac{dZ(x_1)/\tau}{dx_1} \qquad (72)$$

The computational program proceeds by first calculating the location of the point x^* on the airfoil surface at which sonic velocity occurs. This is found from equation (12) as the point where

$$\frac{dG(x)}{dx} = 0$$

For the class of airfoils considered, the function $G(x)$ has one maximum and no minimum in the range 0 to c so that x^* is located approximately at first by evaluating $G(x)$ at points successively farther from the origin until that function passes through its maximum and then more accurately by using a parabolic interpolation. The integral involved in equation (72) is computed by using Simpson's rule. Next, the integral $H(x)$ where

$$H(x) = \frac{2}{\pi} \int_{x^*}^{x} \left[\frac{dG(x_1)}{dx_1}\right]^2 dx_1 \qquad (73)$$

is calculated as a function of x again using Simpson's rule, and then $\overline{C}_p$ and C_p are found by using equations (11) and (3), respectively. Care must be exercised in equation (11) to insure the proper sign is maintained for $x < x^*$ and $x^* < x$. Thus,

$$\overline{C}_p = 2\left(\xi_\infty + \left\{\frac{2}{\pi}\int_{x}^{x^*} \left[\frac{dG(x_1)}{dx_1}\right]^2 dx_1\right\}^{1/3}\right), \quad x < x^* \qquad (74)$$

$$\overline{C}_p = 2\left(\xi_\infty - \left\{\frac{2}{\pi}\int_{x^*}^{x} \left[\frac{dG(x_1)}{dx_1}\right]^2 dx_1\right\}^{1/3}\right), \quad x^* < x \qquad (75)$$

Critical Mach numbers.- The calculation of the lower critical Mach number proceeds by first locating the point on the airfoil surface where C_{p_i}/τ is a minimum and then calculating that minimum value (C_{p_i}/τ). For the airfoils described by equations (6) and (8), C_{p_i}/τ has one minimum along the chord, so that point can be readily calculated by using a technique analogous to the one used for locating x^* in the transonic program. The lower critical Mach number is then found by solving equation (16) iteratively until $M_{cr,\ell}$ is calculated to within a desired accuracy.

The upper critical Mach number is found by calculating the slope of the airfoil surface at the nose from either of equations (6) or (8), inserting the result in equation (19) and iterating on that equation until $M_{cr,u}$ is obtained within a specified accuracy.

Axisymmetric Flows

Surface pressures.- The programs for calculating the surface pressure distribution on slender bodies of revolution by the method of local linearization differ essentially from those developed for two-dimensional flows in that a differential equation must be integrated in all cases for the former. All of the axisymmetric bodies or, in the case of nonaxisymmetric bodies, the "equivalent" bodies of revolution considered here have profiles described by the equations

$$\frac{R}{\ell} = \frac{\tau n^{n/(n-1)}}{2(n-1)}\left[\frac{x}{\ell} - \left(\frac{x}{\ell}\right)^n\right] \tag{76}$$

$$\left(\frac{x}{\ell}\right)_{R_{max}} = \left(\frac{1}{n}\right)^{1/(n-1)} \tag{77}$$

$$\frac{R}{\ell} = \frac{\tau n^{n/(n-1)}}{2(n-1)}\left[1 - \frac{x}{\ell} - \left(1 - \frac{x}{\ell}\right)^n\right] \tag{78}$$

$$\left(\frac{x}{\ell}\right)_{R_{max}} = 1 - \left(\frac{1}{n}\right)^{1/(n-1)} \tag{79}$$

with n = constant ≥ 2. Thus, analogous to the two-dimensional programs, the first step in all of the axisymmetric programs is the calculation of the exponent n according to the relation in either equation (77) or (79).

For purely subsonic flows, the longitudinal perturbation velocity component u on the body surface is calculated by integrating equation (23). Since, however, part of the right-hand side of that equation contains a perfect differential, it is convenient to rewrite the differential equation in terms of another dependent variable in order to avoid unnecessary numerical computation involving that term. If we define

$$f_E(x) = \frac{S''(x)}{4\pi} \ln \frac{S(x)}{4\pi x(\ell - x)} + \frac{1}{4\pi} \int_0^\ell \frac{S''(x) - S''(\xi)}{|x - \xi|} d\xi \tag{80}$$

and

$$V_E(x) = u(x) - f_E(x) \tag{81}$$

then the differential equation for V_E is

$$\frac{dV_E}{dx} = \frac{S'''(x)}{4\pi} \ln \left[1 - M_\infty^2 - k(V_E + f_E)\right] \tag{82}$$

The initial condition becomes, simply

$$V_E(x) = 0 \quad \text{at} \quad S''(x) = 0 \tag{83}$$

Thus, the computational program first integrates the differential equation (82) from the starting condition (83) up to some point x and then calculates $u(x)$ from equation (81) and the pressure coefficient according to equation (21) with $\alpha = 0$. The calculation proceeds first from the starting point to a specified point close to the nose, then returns to the starting point and proceeds to integrate toward the tail. The calculations cannot be carried right to the nose or tail of these sharp-tipped bodies as the local linearization method predicts a logarithmic singularity at these locations, much like that indicated by linearized subsonic flow theory. If at any point along the body the quantity

$$\lambda_E = 1 - M_\infty^2 - M_\infty^2(\gamma + 1) \frac{u}{U_\infty} \tag{84}$$

becomes negative the free-stream Mach number is greater than the lower critical Mach number; the flow field would have a region of local supersonic flow and the above computational program does not apply.

For purely supersonic flows, the computation of u and C_p on the body surface is carried out in a manner similar to that used for purely subsonic flows. In this instance, we define

$$f_H(x) = \frac{S''(x)}{4\pi} \ln \frac{S(x)}{4\pi x^2} + \frac{1}{2\pi} \int_0^x \frac{S''(x) - S''(\xi)}{x - \xi} d\xi \tag{85}$$

$$V_H(x) = u(x) - f_H(x) \tag{86}$$

The differential equation (24) then becomes

$$\frac{dV_H}{dx} = \frac{S'''(x)}{4\pi} \ln \left[M_\infty^2 - 1 + k\left(V_H + f_H\right)\right] \tag{87}$$

subject to the boundary condition

$$V_H(x) = 0 \quad \text{at} \quad S''(x) = 0 \tag{88}$$

The computational program begins the integration with the starting conditions indicated by equation (87) and proceeds toward the nose. In this case, the integration can be carried right to that point because the method of local linearization, consistent with linearized supersonic theory, predicts no singularity in u there for a sharp-tipped body. If, however, before the nose is reached, the quantity

$$\lambda_H = M_\infty^2 - 1 + M_\infty^2(\gamma + 1) \frac{u}{U_\infty} \tag{89}$$

becomes negative, the free-stream Mach number is less than the upper critical. Under these conditions, for the same reason as discussed in connection with supersonic flow over airfoils, a region of subsonic flow would exist in that vicinity and the present program would not be applicable. If λ_E remains positive right up to the nose, the program returns to the starting point and begins to integrate toward the tail. However, a unique feature of supersonic axisymmetric flow causes the flow field to become subsonic near the tail of these bodies and many other classes of shapes that terminate like a cone. This is caused by expansion waves, which emanate from the body surface, reflecting from the bow shock as compression waves and increasing in strength as they coalesce or focus on

the tapering aft section of the body. These incoming compression waves are of sufficient strength to cause a pressure recovery to subsonic speeds upstream of the pointed tail. The basic assumptions of purely supersonic flow are violated, therefore, and further considerations concerning the nature of the transition from supersonic to subsonic flow are necessary if the calculation is to continue to the end of the body. Similar phenomena occur for sonic or near sonic free-stream flow about these bodies and this point is discussed further in that section. The entire matter, however, is not of prime importance since, as pointed out in reference 7, (a) the region involved is generally quite small, (b) viscous effects are certainly present in these regions, and (c) the information is often not required as bodies used in practice or studied in wind-tunnel tests are usually cut off forward of this region. In any case, when this point is reached, the above computational program terminates.

For accelerating transonic flows with $M_\infty \approx 1$, the substitutions used to rewrite the original differential equations for subsonic or supersonic flows do not apply, and it is more convenient to integrate the original differential equation (25) as it stands. Because that equation is singular at the point where $S''(x)$ vanishes, it is necessary to use the Taylor series expansion of the analytic solution for u about that point in order to extend the starting point of the numerical integration far enough away from the singularity so that the standard integrating package can take over. For order-of-error consistency with the integrating scheme, the first six terms of the Taylor series for u are used. The integration is first carried to some specified point close to the nose since, as in the subsonic case, the local linearization method predicts a logarithmic singularity there for a sharp-tipped body. The integration then returns to the starting point, and begins the integration to the tail by again using the Taylor series expansion. However, because the differential equation (25) was derived for accelerating transonic flows, it cannot be used over the entire aft section of the body. The reason for this is similar to that discussed in connection with purely supersonic flows. In this case, the expansion waves that emanate from the body surface are reflected from the sonic line as compression waves; these reflected waves ultimately cause the flow to decelerate somewhere on the aft section of these bodies so that, if information is desired about the flow in those regions, additional assumptions and considerations must be introduced.

Spreiter and Alksne have shown in reference 7 that a reasonable approximation, which is in good accord with experimental evidence, is to join the parabolic differential equation (25) to the hyperbolic differential equation (24) at the point where both u and du/dx as predicted by these two equations match. In the computational program, this is accomplished by calculating the value of du/dx from the hyperbolic equation (24) by using the value of u predicted by the parabolic equation (25) as the latter is integrated toward the tail. When the values of du/dx predicted by both equations match, a transfer is made to the hyperbolic equation and the calculation continues toward the tail. As was the case for purely supersonic flows, for sonic or near sonic free-stream flows the region of supersonic flow does not extend all the way to the rear tip of bodies such as those considered here that terminate like a cone. Rather, the flow decelerates and reaches sonic velocity at a point somewhat forward of the rear tip. While the flow in the immediate vicinity of the rear tip is known to be subsonic, little more has been established about the nature of the flow in that region. It is clear, however, that two possibilities exist regarding the nature of the transition from supersonic to subsonic along the body surface. The transition may be accomplished in a discontinuous manner involving one or more shock waves that extend to the body surface, or it may be accomplished in a continuous manner with a smooth deceleration through sonic velocity. In either case, shock waves are present in the flow field but do not extend to the body surface for the latter case of smooth deceleration through sonic velocity. It is assumed in the present analysis that this transition from supersonic to subsonic flow along the body surface is continuous and that the results predicted by the hyperbolic equation (24) are valid up to the immediate vicinity of the rear sonic point. Furthermore, it is assumed that the results for subsonic flow along the remainder of the body can be calculated by means of the elliptic equation (23). Although both equations (23) and (24) predict a logarithmically infinite slope at the rear sonic point, the extent of the region involved is exponentially small, so that Spreiter and Alksne (ref. 7) were able to show by the method of isoclines that the transfer from the hyperbolic to the elliptic equation involves a discontinuity that is restricted to a region sufficiently small as to cause no numerical difficulty. Consequently, it is possible to carry the calculation through the sonic point and effect the transfer of equations by means

of a linear interpolation, subsequently continuing the calculation toward the rear tip by using the elliptic equation (23). The calculation then proceeds to a designated point near the rear tip and terminates.

In each of the preceding computational programs, all differential equations are integrated by using Hamming's modified predictor-corrector method described in the Scientific Subroutine Package (SSP) available from the IBM Corporation. The integrals involved in those differential equations are evaluated numerically by Simpson's rule while any derivatives which must be calculated numerically are found by using a central 7-point difference formula. Analytic expressions are provided for the area distribution function and its derivatives although the programs can be easily modified to accommodate tabular data of this form, as would be the case for arbitrary slender bodies of revolution.

Flow-field pressures.- Calculation of the flow-field pressure distribution for the axisymmetric flow about slender bodies of revolution is readily obtained through use of the equivalence rule to extend the results for the surface pressures out into the flow field. The result of this application, given in equation (66), indicates that the pressure coefficient at a point removed by a lateral distance r from the body axis is found by calculating the function f_1 where

$$f_1(x,r) = -\frac{S''(x)}{\pi} \ln\left(\frac{r}{R_{eb}}\right) - \left(\frac{R_{eb}}{r}\right)^2\left(\frac{dR_{eb}}{dx}\right) \tag{90}$$

and adding to it $-2u_B/U_\infty$ where, depending upon the Mach number range, u_B is calculated from the subsonic, supersonic, or $M_\infty \approx 1$ programs discussed above. Consequently, the flow-field calculation can be accomplished by modifying only the output subroutines of these programs to include this additional calculation if it is required.

Nonaxisymmetric Flows

Surface pressures - nonlifting and lifting.- Calculation of the surface pressures on nonlifting parabolic-arc bodies having elliptical cross sections that maintain a constant ratio of major to minor axes along the body length is also conveniently accomplished through use of the equivalence rule. Equation (50), which is the result of application of this rule to that class of shapes, indicates the pressure coefficient

at any point on those bodies may be found by adding the function g_1 where

$$g_1(x,\theta;\lambda) = -\frac{S''(x)}{2\pi} \ln\left[\frac{(\lambda+1)^2}{4\lambda}\right] - \left(\frac{dR_{eb}}{dx}\right)^2 \left(2 - \lambda \frac{\cos^2\theta + \lambda^2 \sin^2\theta}{\cos^2\theta + \lambda^4 \sin^2\theta}\right) \tag{91}$$

to $-2u_B/U_\infty$, where u_B is calculated from the appropriate (subsonic, supersonic, or $M_\infty \approx 1$) local linearization program for axisymmetric flow past a parabolic-arc body of revolution.

For lifting flows about these bodies at small angles of attack, equation (55) indicates that in addition to the function g_1 another function g_2 must be added to $-2u_B/U_\infty$ to obtain the surface pressure coefficient where

$$g_2(x,\theta;\lambda,\alpha) = -2\alpha \left(\frac{dR_{eb}}{dx}\right) \frac{\lambda^{5/2}(\lambda+1)(\cos^2\theta + \lambda^2 \sin^2\theta)^{1/2} \sin\theta}{(\cos^2\theta + \lambda^4 \sin^2\theta)} - \alpha^2 \left[\frac{-\lambda \sin^2\theta + (\lambda^2 + 2\lambda)\cos^2\theta}{\cos^2\theta + \lambda^4 \sin^2\theta}\right] \tag{92}$$

Consequently, to obtain the surface pressure distributions on these bodies for nonlifting and lifting situations, it is necessary to alter only the output subroutines of the subsonic, supersonic, and $M_\infty \approx 1$ local linearization programs for axisymmetric flow about the equivalent parabolic-arc body of revolution by adding the functions g_1 and g_2.

Flow-field pressures - nonlifting.- The flow-field pressure distribution for nonlifting flow about this class of bodies is found in a similar manner. Application of the transonic equivalence rule indicates that the pressure coefficient at an arbitrary point $re^{i\theta}$ in the crossflow plane can be found by addition of the function

$$g_3(x,r,\theta;\lambda) = \text{R.P.}\left(-\frac{S''(x)}{\pi} \ln\left\{\frac{1}{2\sqrt{\lambda}}\left[\frac{r}{b} e^{i\theta} + \left(\frac{r^2}{b^2} e^{2i\theta} - \lambda^2 + 1\right)^{1/2}\right]\right\}\right)$$

(eq. (93) cont. on next page)

$$+\left(\frac{dR_{eb}}{dx}\right)^2\left\{\frac{2(\lambda^2-1)}{\left[\frac{r}{b}e^{i\theta}+\left(\frac{r^2}{b^2}e^{2i\theta}-\lambda^2+1\right)^{1/2}\right]\left[\frac{r^2}{b^2}e^{2i\theta}-\lambda^2+1\right]^{1/2}}\right.$$

$$\left.\left.-\frac{\lambda}{\left[\frac{r^4}{b^4}+(\lambda^2-1)^2\right]^{1/2}}\right\}\right) \tag{93}$$

to $-2u_B/U_\infty$ where again u_B is calculated from the appropriate local linearization program for axisymmetric flow about the equivalent parabolic-arc body of revolution.

Critical Mach numbers.- The programs for calculating the critical Mach numbers for axisymmetric flow about the bodies of revolution described by equations (76) and (78) primarily differ from the corresponding two-dimensional programs in that, for the latter, integral relationships exist between the critical Mach numbers and those values of u/U_∞ that determine them (see eqs. (16) and (19)), whereas for the axisymmetric cases, the analogous relationships are through different equations. Specifically, the calculative program that determines the lower critical Mach number on these bodies of revolution integrates differential equation (23) past the point of maximum u/U_∞ by estimating an initial value for $M_{cr,\ell}$. If the integration goes through that point with the quantity

$$\lambda_E = 1 - M_\infty^2 - M_\infty^2(\gamma+1)\left(\frac{u}{U_\infty}\right)_{max} \tag{94}$$

greater than zero, then the value of the free-stream Mach number is systematically increased and the integration repeated with the new value. If, on the other hand, λ_E becomes negative before u/U_∞ reaches its maximum, the value of M_∞ is reduced and the integration again repeated. The iteration continues until λ_E simultaneously vanishes at the point where u/U_∞ reaches its maximum and the critical Mach number is determined to within a given tolerance. We note that at the point where λ_E vanishes, equation (23) predicts a logarithmically infinite slope. This fact causes no numerical difficulty, however, as the singularity is of such a weak type that the lower critical Mach number can be calculated to within any reasonable tolerance. For the bodies considered herein, the point of

maximum u/U_∞ always lies aft of the point where $S''(x)$ vanishes, so it is unnecessary to carry the integration from that starting point to the nose.

The program for calculating the upper critical Mach number proceeds by estimating an initial value of $M_{cr,u}$ and then integrating differential equation (24) toward the nose. If the integration reaches the nose with the quantity

$$\lambda_H = M_\infty^2 - 1 + M_\infty^2(\gamma + 1)\left(\frac{u}{U_\infty}\right)\Bigg|_{x=0} \tag{95}$$

greater than zero, then the value of the free-stream Mach number is systematically reduced and the integration repeated with that Mach number. If λ_H becomes negative before the nose is reached, the value of M_∞ is increased and the integration repeated. The iteration continues until λ_H vanishes at the nose and the upper critical Mach number is determined to within a specified tolerance. As with the calculation of the lower critical Mach number, the logarithmic singularity in slope in equation (24) implied by the vanishing of λ_H at the nose causes no computational difficulty.

RESULTS AND DISCUSSION

Two-Dimensional Flow

Theoretical surface pressure distributions calculated for members of the class of airfoils described by equations (6) and (8) having positions of the point of maximum thickness at 30, 40, 50, 60, and 70 percent of the chord are presented in figure 2 for free-stream Mach number $M_\infty = 1$ and also at the lower and upper critical values of the similarity parameter ξ_∞ of the airfoil in question. The experimental results in that figure are for $M_\infty = 1$ from reference 26 and were obtained for a family of airfoils having thickness ratios τ ranging from 6 to 12 percent.

The theoretical results at $M_\infty = 1$ for these have been given previously in reference 19; however, in that work, the results presented for the airfoils with maximum thickness at 30 and 40 percent are not exact in that the actual equations describing the airfoil ordinates are approximated in order to obtain analytical solutions. Thus the results

given in figure 2 for these two airfoils are new and serve to complete the theoretical and experimental comparisons for the entire series of airfoils tested.

We note the remarkably large extent of the transonic range for these airfoils as indicated in the table below by the values of the critical Mach numbers for members of that class having a thickness ratio $\tau = 1/12$.

$\left(\frac{x}{c}\right)_{Z_{max}}$	.30	.40	.50	.60	.70
$M_{cr,u}$	2.00	1.66	1.49	1.31	1.24
$M_{cr,\ell}$	0.75	0.77	0.79	0.77	0.75

Moreover, these ranges would be even greater were the airfoils lifting. These same results are presented in figure 3, which shows the variation of the critical Mach numbers as a function of the location of the point of maximum thickness; this plot provides an insight into the relative size of the Mach number ranges over which the theory and calculative programs are applicable. Corresponding results, which are discussed later, are shown for bodies of revolution having profiles similar to those of the airfoils. The vertical distance between the upper and lower curves indicates the extent of the transonic range for these bodies and the range in which the purely subsonic or purely supersonic theories are inappropriate. The slight discontinuity that appears in the slopes of the curves at

$$\left[\left(\frac{x}{c}\right)_{Z_{max}}, \left(\frac{x}{\ell}\right)_{R_{max}}\right] = 0.5$$

occurs because of the two different expressions used to describe the bodies with maximum thickness forward or rearward of the midpoint. The figure clearly demonstrates the rapid variation in the upper critical Mach number $M_{cr,u}$ as the point of maximum thickness moves toward the nose. It indicates, furthermore, that in view of the large supersonic values predicted for $M_{cr,u}$ for these blunter-nosed shapes, the assumptions inherent in the small-disturbance approximation are beginning to break down and that

the theory should be applied with considerable caution to those shapes having their maximum thickness point close to the nose.

With regard to the theoretical and experimental pressure distributions at $M_\infty = 1$ ($\xi_\infty = 0$) in figure 2, it is apparent that except for the immediate vicinity of the trailing edge, these comparisons indicate essential agreement between the calculated and measured pressure distributions. The general trend of the theory to agree better with the data for thicker airfoils and for airfoils with maximum thickness forward has been examined in some detail previously (ref. 18), and found to be consistent with the anticipated effects of wind-tunnel-wall interference and of the viscous boundary layer.

With respect to the differences that occur near the trailing edge, the question arises whether they are primarily the result of boundary-layer shock-wave interaction, as most would suppose, or whether they might be inherent in the inviscid solution. The latter would be the case if the local Mach number immediately upstream of the trailing edge were not high enough for an oblique shock wave to turn the flow through the required angle. If that should occur, the shock wave would detach from the trailing edge and move forward onto the aft part of the airfoil, substantially altering the pressure distribution there.

To investigate this possibility, we must consider the following expression (ref. 27) for the transonic approximation for the shock polar, where u and w refer to the Cartesian components of the perturbation velocity and subscripts a and b refer to conditions ahead and behind the shock wave:

$$(1 - M_\infty^2)(u_a - u_b)^2 + (w_a - w_b)^2 = \frac{M_\infty^2(\gamma + 1)}{U_\infty}\left(\frac{u_a + u_b}{2}\right)(u_a - u_b)^2 \qquad (96)$$

Upon introducing $C_p = -2u/U_\infty$ and $dZ/dx = w/U_\infty$, where U_∞ refers to the free-stream velocity, and setting $M_\infty = 1$, $(dZ/dx)_b = 0$, and expressing the relations in terms of $\overline{C}_p$ rather than C_p, equation (96) becomes

$$\left[\frac{d(Z/\tau)}{dx}\right]_a^2 = -\frac{1}{16}(\overline{C}_{p_a} + \overline{C}_{p_b})(\overline{C}_{p_b} - \overline{C}_{p_a})^2 \qquad (97)$$

With $\left[d(Z/\tau)/dx\right]_a$ known from the geometry of the airfoil, and $\bar{C}_{p_a}$ assumed to be as indicated by the local linearization theory, this relation may be solved for $\bar{C}_{p_b}$, the reduced pressure coefficient immediately downstream of the trailing edge, provided $-\bar{C}_{p_a}$ is sufficiently large for the shock wave to be attached to the trailing edge. If $-\bar{C}_{p_a}$ is less than a limiting value $(-\bar{C}_{p_a})_{att}$, which depends on $\left[d(Z/\tau)/dx\right]_a$, it is not possible for an oblique shock wave to turn the flow the required angle. The appropriate expression for $(-\bar{C}_{p_a})_{att}$ may be determined by solving

$$\frac{d}{d\bar{C}_{p_b}}\left[\frac{d(Z/\tau)}{dx}\right]_a^2 = -\frac{1}{16}(\bar{C}_{p_b} - \bar{C}_{p_a})(3\bar{C}_{p_b} + \bar{C}_{p_a}) = 0 \qquad (98)$$

for $\bar{C}_{p_b}$ in terms of $\bar{C}_{p_a}$ to obtain $\bar{C}_{p_b} = -\bar{C}_{p_a}/3$, and substituting into equation (97) to obtain

$$(\bar{C}_{p_a})_{att} = -\frac{3}{2^{1/3}}\left[\frac{d(Z/\tau)}{dx}\right]_a^{2/3} \qquad (99)$$

A necessary condition for the shock wave to be attached to the trailing edge of the airfoil is thus that $\bar{C}_{p_a} \leq (\bar{C}_{p_a})_{att}$. It may be seen from the values listed for $\bar{C}_{p_a}$ and $(\bar{C}_{p_a})_{att}$ in Table I that this condition is satisfied for each of the airfoils for which results are shown in figure 2.

TABLE I

SIGNIFICANT VALUES FOR $\bar{C}_p$ AND SLOPE AT THE TRAILING EDGE OF THE AIRFOILS PRESENTED IN FIGURE 1

$(x/c)_{Z_{max}}$	$\left[\frac{d(Z/\tau)}{dx}\right]_a$	$\bar{C}_{p_a}$	$(\bar{C}_{p_a})_{att}$	$\bar{C}^*_{p_a}$	$(\bar{C}_{p_b})_{weak}$	$(\bar{C}_{p_b})_{strong}$
.30	-0.86	-4.04	-2.15	-2.27	-2.73	3.86
.40	-1.18	-3.96	-2.67	-2.82	-2.03	3.57
.50	-2.00	-4.45	-3.78	-4.00	-1.04	3.42
.60	-2.82	-5.23	-4.75	-5.03	-0.53	0.25
.70	-4.32	-6.61	-6.32	-6.68	0.25	3.91

The margin by which this condition is exceeded diminishes rapidly as the maximum thickness moves aft on the airfoil, however, and it seems likely that the condition for an attached shock wave may not be satisfied for airfoils of this family if the maximum thickness were much further aft than the most extreme of the cases considered, namely 0.70 chord.

A further condition required for the surface pressure distribution to be independent of conditions downstream of the airfoil in the absence of a boundary layer is that the flow be supersonic downstream of the trailing shock wave. Now, it is a well-known property of the transonic shock polar that can be seen from equation (97), or the listings of Table I, that $\bar{c}_{p_b}$ may be either positive or negative, but that $\bar{c}_{p_a} < \bar{c}_{p_b} \leq -\bar{c}_{p_a}$ if attention is confined to physically significant compressive shock waves described by real solutions of equation (97). Of the two solutions of that equation that satisfy this condition, one is always positive, indicating subsonic flow, and the other may be positive or negative, indicating either subsonic or supersonic flow. To further illustrate this point, the two values for $\bar{c}_{p_b}$ are indicated by solid and open half circles for each of the airfoils on figure 2. The limiting value for $\bar{c}_{p_a}$ for which the flow is exactly sonic immediately downstream of a shock wave attached to the trailing edge is designated by $\bar{c}^*_{p_a}$ and may be determined from equation (97) by setting $\bar{c}_{p_b}$ equal to zero. It is

$$\bar{c}^*_{p_a} = -2^{4/3} \left[\frac{d(Z/\tau)}{dx}\right]^{2/3}_a \tag{100}$$

Values for $\bar{c}^*_{p_a}$ are listed in Table I for each of the airfoils for which results are shown in figure 2.

If $\bar{c}^*_{p_a} \leq \bar{c}_{p_a} \leq (\bar{c}_{p_a})_{att}$, a shock wave may be attached to the trailing edge, but the flow immediately downstream of it must be subsonic. Under these circumstances, which prevail for the airfoils of figure 2 with maximum thickness at 0.70 chord, effects arising in the wake and elsewhere downstream of the airfoil inevitably influence conditions at the trailing edge. This result opens the possibility that these influences might be sufficient, under certain circumstances, to cause the trailing shock wave to move forward onto the airfoil, and thereby to

to produce at least part of the discrepancies apparent in figure 2 between the theoretical and experimental pressure distributions near the trailing edge.

If $\overline{c}_{p_a} < \overline{c}^*_{p_a} < (\overline{c}_{p_a})_{att}$, as it is for the remainder of the airfoils considered in figure 2, the shock wave may be attached to the trailing edge, but the flow immediately downstream of it may be either subsonic or supersonic according to whether the shock wave is strong or weak. If the shock wave is of the strong type, the situation would be substantially the same as just described; and the sources of the differences between the theoretical and experimental pressure distributions near the trailing edge might be sought in either inviscid or viscous phenomena. If, on the other hand, the shock wave is of the weak type, the flow is supersonic immediately downstream of the shock wave, and no effects of downstream origin can influence conditions at the airfoil in the absence of viscosity. To the extent that this prevails, the above results support the general belief that the differences between theoretical and experimental pressure distributions over the aft portion of airfoils with small or moderate trailing-edge angles must involve viscous processes, in particular, effects of shock-wave boundary-layer interaction. While the inviscid theory is capable of providing a good approximation for the pressure distribution over most of the airfoil, the nature and magnitude of the deficiencies near the trailing edge indicate that an improved theory, which includes viscous effects, is required to provide satisfactory predictions of drag at $M_\infty = 1$.

Axisymmetric Flows

Surface pressures.- Experimental surface pressure distributions from references 35 and 36 together with the theoretical results calculated using the local linearization method are presented in the lower plots of figure 4, (a) through (e), for members of the class of bodies described by equations (77) and (79) having a thickness ratio $D/\ell = 1/12$ and locations of the point of maximum thickness at 30, 40, 50, 60, and 70 percent of the body length. The theoretical results for the bodies with maximum diameter at 30, 50, and 70 percent of the body length have been given previously (refs. 7 and 10); those for the bodies with maximum diameter at 40 and 60 percent of the body length are new and thus complete the comparisons for the entire series of bodies tested. Also included on

the plots for the bodies with maximum diameter at 30, 50, and 70 percent of the body length are a second set of experimental results (ref. 10) indicated by closed circles. These results were obtained when the same models used in the original tests in the 14-foot transonic wind tunnel were tested in the Ames 12-foot pressure wind tunnel with solid walls under choked conditions to simulate flow with $M_\infty = 1$. That this provides, under certain restrictions, conditions that bear close resemblance to an unbounded flow at $M_\infty = 1$ has been verified by numerous investigations (refs. 10 and 37). It may be seen that the pressure distributions calculated using the local linearization method are in essential agreement with the measurements in both wind tunnels over the forebodies, but that substantial discrepancies appear among the results for the afterbodies, particularly for the bodies with maximum thickness forward of the midpoint. As noted originally (ref. 10), the data from the choked wind tunnel are generally on the opposite side of the theoretical curve from the data from the transonic wind tunnel.

With respect to the discrepancies on the afterbodies, many would dismiss further discussion by attributing the differences to shock-wave boundary-layer interaction effects not included in the theory. While there is little doubt that such effects are important, the characteristic diagram given in figure 5 suggests that the discrepancies are due, at least in part, to wind-tunnel-wall interference. This diagram shows the characteristic lines for an unbounded flow with $M_\infty = 1$ past a parabolic-arc body of revolution with $D/\ell = 1/12$. They have been calculated by applying the transonic similarity rule for axisymmetric flow (ref. 38) to a related diagram (ref. 39) calculated for a parabolic-arc body of revolution with $D/\ell = 1/6$. The position of the wall with respect to a 6-inch diameter model in the 12-foot pressure wind tunnel is as indicated, and the nearest part of the wall in the tests in the 14-foot transonic wind tunnel is 7/6 as far away. Although it was thought at the time the tests in the 14-foot transonic wind tunnel were conducted that the 6-inch diameter of the models was sufficiently small to avoid significant effects of wind-tunnel-wall interference, this diagram shows that such may not be the case because Mach waves originating from the forepart of the body are indicated to be reflected from the walls onto the aft part of the body. It can be seen, furthermore, that the most upstream reflected Mach wave strikes the body at about $x/\ell = 0.6$ in the 12-foot wind tunnel,

and slightly aft of that location in the 14-foot wind tunnel. The effect of the reflected waves striking the body is to make the pressure coefficients more negative in the 12-foot wind tunnel, because the outgoing characteristics represent expansion waves that reflect from the solid wall of the tunnel as rarefaction waves. The effects are amplified, moreover, because of the focusing properties of the reflected axisymmetric waves as they collapse down onto a part of the body that has a smaller circumference than that from which they originated. The sign of the corresponding effects in the 14-foot transonic wind tunnel is not so simple to ascertain, since the reflections from the partly open wall of that wind tunnel are very nearly equal in magnitude, but opposite in sign, to that of the reflections from the solid wall of the 12-foot wind tunnel. In addition to the direct effects of the reflected waves impinging on the rear of the body, there exists the distinct possibility of significant augmentation arising from the interaction of the boundary layer with a shock wave that may form adjacent to the body. The latter may form either because of coalescence of compression waves reflecting from the body or because of boundary-layer separation resulting from the wall-induced steepening of the adverse pressure gradients. In either case, it is clear that considerable additional study will have to be made before it is possible to properly evaluate the significance of discrepancies between theoretical and experimental pressure distributions on the aft parts of slender bodies of revolution. For these same bodies, the upper and lower critical Mach numbers have been calculated and are presented in figure 3 as a function of the location of maximum thickness. In comparison with the results shown for airfoils having the same profiles and thickness ratio, the critical Mach numbers for the bodies of revolution exhibit the same general variation but are substantially closer to one. This is in keeping with the concept that a body of revolution produces a much smaller disturbance to the flow than an airfoil of infinite span. Analogous to the two-dimensional case, moreover, the axisymmetric results indicate that, because of the large supersonic values of $M_{cr,u}$ predicted for those bodies having their maximum thickness location close to the nose, the theory should be applied with considerable caution to those shapes.

Flow-field pressures.- In addition to the surface pressure distributions on these bodies, the pressure distributions at various locations in the near flow field have been calculated and are given in the upper

plots of figure 4, (a) through (e). The flow field data presented in those plots are from the Ames 14-foot transonic wind tunnel and were measured along lines parallel to the body axis but removed from it by distances of 1, 2, and 4 times the maximum diameter D of the body. The theoretical results were calculated by using equation (63) together with the local linearization results for u_B/U_∞ provided by integrating equation (25). Except near the rear of the bodies where effects associated with the discrepancies discussed above for the surface pressures are in evidence, the theoretical and experimental results are in satisfactory agreement for r/D as large as 4 for the 50-, 60-, and 70-percent bodies, and at least 2 for the 30- and 40-percent bodies. Deterioration sets in with increasing r/D for any body because of the growing violation of the small r approximation of the theory, and is greatest for the 30-percent body because of the greater strain imposed on the slender-body approximation by the blunter nose of this body. Altogether, the comparisons indicate a wide region of applicability of the theory, particularly when considered with respect to possible applications of the transonic equivalence rule to configurations having wings or related extremities of such size that they, rather than the body, provide the major contribution to C_p at lateral distances of the order of those for which results are shown in figure 4.

For two of those bodies studied above, corresponding comparisons of surface and flow-field pressure distributions for purely subsonic and purely supersonic flows are presented in figures 6 and 7. Figure 6 exhibits the results for free-stream Mach number $M_\infty = 0.90$ for the 30- and 70-percent bodies. With the exception of the previously discussed discrepancies near the rear of the bodies, agreement between experiment and theory is quite satisfactory. The calculated and measured surface pressure distributions compare quite well for both bodies, while the agreement for the flow-field pressures for the 30-percent body is reasonable even out to $r/D = 4$.

The flow-field comparisons for the 70-percent body are good up to about $x/\ell = 0.6$, at which point the theoretical results become somewhat more negative than the data. Although we cannot conclusively confirm it, it seems reasonable to expect that for this body the discrepancy may be caused by separation of the boundary-layer just beyond the point of maximum thickness. This would have the effect of making the part of the

body aft of the separation point appear cylindrical to the outer flow and, as a consequence, would tend to increase the pressure level in that region. In figure 7, the analogous theoretical and experimental results for purely supersonic flow at $M_\infty = 1.20$ are presented for the 70-percent body. Similar results for the 30-percent body are not given since no experimental results are available at free-stream Mach numbers greater than the upper critical for this body; the highest test Mach number reported is 1.20, whereas the upper critical for the 30-percent body having $D/\ell = 1/12$ is 1.68. For the 70-percent body, we note that agreement between predicted and measured surface pressure distribution is good right up to the immediate vicinity of the sting. This is reasonable since in this case the absence of upstream influence should minimize the effect of the sting on the flow ahead of it. The flow-field comparisons indicate that the predicted pressure is somewhat higher than the measured results on the forebody and somewhat lower on stations beyond about $x/\ell = 0.6$. This effect is probably caused by not properly accounting for the regions of influence of the flow field. For example, the location of the shock wave, although not of primary importance in calculating the surface pressure distribution, is vital in determining the extent of the region of influence. Beyond that region, i.e., in front of the shock, no disturbances exist, so that it is proper to use the equivalence rule to extend surface pressures only out to the lateral extent of the bow shock. Thus, some of the flow-field results shown in figure 7, particularly on the forebody, have no doubt been extended laterally beyond that point. Consequently, to properly account for flow-field pressure distributions about these bodies, it is necessary first to determine the shock location and then to use the equivalence rule in regions within that domain.

Nonaxisymmetric Flows

Nonlifting bodies - surface and flow-field pressures.- As the first application of the transonic equivalence rule to nonaxisymmetric flows, we consider a family of bodies with elliptic cross sections tested in the Ames 14-foot transonic wind tunnel. Figure 8 presents the surface pressure distributions measured along the extremities of the major and minor axes ($\theta = 0^\circ$ and 90°) at $M_\infty = 1$ for three bodies having values of 1.5, 2.0, and 3.0 for the ratio $\lambda = a/b$ of major to minor axes of the elliptic cross section, and having the same longitudinal distribution of cross

section area $S = \pi ab$ as the parabolic-arc body of revolution with $D/\ell = 1/12$ for which results are shown in figure 4(c). Also included in figure 8 are the corresponding pressure distributions calculated by using the equivalence rule via equation (50) together with the result provided by the local linearization method from equation (22) for u_B/U_∞ on the equivalent parabolic-arc body. As in the previous comparisons, the theoretical and experimental results are in good agreement except near the rear of the bodies, where at least part of the discrepancies must be attributed to the extraneous effects of the wind-tunnel walls. Perhaps the most striking feature of the results shown on figure 8 is the smallness of the effects of the ellipticity of the cross section.

The corresponding results for the flow field are shown in figure 9 for the body having $\lambda = 3$. As may be anticipated from the results of figure 8 for the surface pressures, the value for C_p at a given x are virtually independent of the azimuthal angle θ, and are, furthermore, very nearly identical to those shown in figure 4(c) at the same r/D for the equivalent parabolic-arc body of revolution. For these reasons, the flow field results for the other two bodies of this series having smaller λ have been omitted.

Analogous comparisons of theoretical and measured surface and flow-field distributions for purely subsonic and purely supersonic flow are presented in figures 10 and 11 for the body having $\lambda = 3$. Figure 10 exhibits the pressure distributions at $M_\infty = 0.90$ along the extremities of the major and minor axes ($\theta = 0^\circ$, 90°) both on the surface and at points at the same angular position but located at lateral distances of $r/D = 1$, 2, and 3 in the flow field; figure 11 presents similar results for $M_\infty = 1.20$ at one angular location ($\theta = 0^\circ$) and one flow-field position ($r/D = 2$) as only these data are available for comparison. As with the case of $M_\infty = 1.00$ flow about this body, the agreement between theory and experiment is quite good for both surface and flow-field pressure distributions. We note again the small effect of the ellipticity of the cross section; at a lateral distance of $r/D = 2$, figure 10 shows that at $M_\infty = 0.90$ the flow field is, for all practical purposes, axisymmetric. Although data are available at only one angular position for the flow at $M_\infty = 1.20$, theoretical results indicate that this weak effect of the ellipticity also persists into the purely supersonic flow regime. In this case, it is interesting to observe with regard to the comparison of

the flow-field results at $r/D = 2$ that at points near the nose, the data are tending toward zero while the theory, of course, displays effects associated with the logarithmic singularity at the nose. As with the results for purely supersonic flow over the body having maximum thickness at 70 percent of its length given in figure 7, this serves to demonstrate again that in regions near the nose, the theoretical results have undoubtedly been overextended into the flow field beyond the bow shock.

Lifting bodies - surface pressures.- As the first application to a lifting body, we have considered flow with $M_\infty = 1$ past the parabolic-arc body of revolution with $D/\ell = 1/12$ for which the results for zero angle of attack are shown in figure 4(c). Experimental results for this body obtained in the Ames 14-foot transonic wind tunnel (ref. 19) at angles of attack $\alpha = 2$, 4, and 6° are shown in figure 12 for five lines along the surface of the body, namely $\theta = 0^\circ$, $\pm 40^\circ$, and $\pm 90^\circ$. Also included on those plots are pressure distributions calculated by using the results of the transonic equivalence rule given by equation (55) with $\lambda = 1$ together with u_B/U_∞ calculated from equation (25). It is evident that agreement is good along most of the length of the body, while significant discrepancies occur near the rear, much as in the case of the nonlifting bodies. The similarity of the comparisons, particularly at the smaller angles of attack, to those for $\alpha = 0^\circ$ shown in figure 4(c) lend support to the idea that wind-tunnel-wall interference effects are significantly affecting the results along the rear of the body.

Similar comparisons of theoretical and experimental results for lifting bodies of elliptic cross section in flow with $M_\infty = 1$ are shown in figures 13, 14, and 15 for bodies having values 1.5, 2.0, and 3.0 for the ratio $\lambda = a/b$ of major to minor axes. All three bodies have the same longitudinal distribution of cross section area as the parabolic-arc body of $D/\ell = 1/12$ for which results are shown in figures 4(c) and 12. The experimental results are from reference 19 and were obtained in tests in the Ames 14-foot transonic wind tunnel, and are thus directly comparable to those presented in figures 4(c) and 12 for the equivalent parabolic-arc body of revolution tested in that wind tunnel.

The comparisons show that the trends already described for the body of revolution are repeated; good agreement occurs along most of the length of each body, and notable deterioration sets in along the rear part. The

variation with θ of the differences between the theoretical and experimental pressures that becomes increasingly evident as α is increased to 6° suggests that additional effects having a nonaxisymmetric character are present along the aft portion of these bodies. Although we cannot conclusively confirm it, we are inclined to conjecture that the experimental results are beginning to be influenced by vortex separation caused by the crossflow velocity component, as is familiar at both subsonic and supersonic speeds. Support to this suggestion is lent by examination of the pressure distribution results for the $\theta = 0^{\circ}$ station on these bodies as both the angle of attack and the ellipticity are increased. Agreement between theory and experiment for that station for the body of revolution (fig. 12) remains satisfactory up to 70 percent of the body length at the smaller angles of attack ($\alpha = 2^{\circ}$, 4°), while at $\alpha = 6^{\circ}$ the data begin to depart from the theory at approximately 65 percent of the body length. This effect becomes more pronounced as the ratio λ of major to minor axes increases, so that for the most elliptic body ($\lambda = 3$), the point at which disagreement occurs has moved up to approximately 45 percent of body length. On the other hand, agreement at the other angular stations ($\theta = \pm 40^{\circ}$, $\pm 90^{\circ}$) remains quite good for all four of these bodies even at $\alpha = 6^{\circ}$. Now if, as the angle of attack is increased, the crossflow is unable to negotiate the curve at the lateral extremities of these bodies ($\theta = 0^{\circ}$, 180°) and separation subsequently occurs somewhere in that vicinity, then experimental pressure distributions for separated flows about bluff objects such as circular cylinders and spheres (ref. 40) indicate a larger pressure would exist at that location than indicated by theory. Since the data display this tendency and, as is characteristic of vortex separation phenomena, because this effect becomes more pronounced as the angle of attack increases, we suspect that a limited region of vortex separation exists on the lee parts of these bodies beginning at approximately the $\theta = 0^{\circ}$ meridian but not extending to the 40° station.

Analogous comparisons of theoretical and measured surface pressure distributions for purely subsonic and purely supersonic flow are given in figure 16 and 17 for the body having $\lambda = 3$. Figure 16 displays results at $M_{\infty} = 0.90$ and figure 17 at $M_{\infty} = 1.20$. The trends that were observed for the $M_{\infty} = 1$ cases are also present for the purely subsonic and supersonic flows. For the $\theta = \pm 40^{\circ}$, $\pm 90^{\circ}$ stations, agreement is good for all three angles of attack over the majority of the body length with

significant discrepancies occurring only near the tail; for the $\theta = 0^{\circ}$ station, the point at which theory and data clearly depart moves from approximately 75 to 45 percent of the body length as the angle of attack is increased to 6°.

Pressure distributions on slender wing-body combinations.- In order to demonstrate how the present results can be extended to predict transonic flows about more complex configurations, we have considered two particular wing-body combinations. It is instructive, however, first to recall the example of the thin cone-cylinder, or equivalently, a cone of finite length, considered theoretically in reference 6, experimentally in reference 41, and summarized in reference 9. In general, the ordinates of the upper surface of the cone are given by

$$Z_u(x,y) = \frac{t\sqrt{m^2x^2 - y^2}}{2m\ell} \tag{101}$$

where m is the tangent of the semiapex angle of the planform, ℓ is the length of the cone, and t is the maximum thickness of the cone. The elliptic section in the plane $x = x_1$ thus has major and minor semiaxes $a = mx_1$ and $b = tx_1/2\ell$, and cross section area $S(x_1) = \pi ab = \pi tmx_1^2/2\ell$. Although the theory was developed generally, the pressure distribution has been measured for only one example, namely that defined by $m = 1/2$ and $t/\ell = 0.06$. If we consider the cone to be a triangular wing, such values correspond to an aspect ratio of 2 and a thickness ratio of 0.06. The equivalent body of revolution is a slender cone with semiapex angle $\omega = (mt/2\ell)^{1/2}$, which in this instance is equal to 0.1225 radians or 7°. Equations (36) and (51) could be used for $\phi_{2,t}$ and $\phi_{2,\alpha}$, but it is much simpler for the calculation of the surface pressure distribution to adopt the usual practice of thin wing theory and evaluate these quantities on the wing surface approximated to be at $z = 0$ and for $-mx \leq y \leq mx$. With the additional neglect of b^2 compared with a^2, $\phi_{2,t}$ and $\phi_{2,\alpha}$ may be approximated by

$$\left.\begin{aligned} \phi_{2,t} &= \frac{U_\infty tmx}{2} \\ \phi_{2,\alpha} &= \pm U_\infty \alpha (m^2x^2 - y^2)^{1/2} \end{aligned}\right\} \tag{102}$$

where the upper sign is to be used on the upper surface and the lower sign on the lower surface of the wing. With the introduction of the further approximation customary in wing theory that the pressure coefficient is given by

$$(C_p)_W = -\frac{2}{U_\infty}\frac{\partial \phi}{\partial x} \tag{103}$$

it was shown in reference 6 that

$$(C_p)_W = (C_p)_B - \frac{mt}{2\ell}\left(1 + \ln\frac{m\ell}{2t}\right) \mp \frac{2\alpha m^2 x}{(m^2x^2 - y^2)^{1/2}} \tag{104}$$

where the convention concerning upper and lower signs still holds, and $(C_p)_B$ represents the pressure distribution on the surface of the equivalent body, a circular cone-cylinder, at zero angle of attack. If the method of local linearization is used to determine the latter, we have that (refs. 7 and 9)

$$(C_p)_B = -2\omega^2 \ln\frac{\omega x}{\ell} + \omega^2 \ln\left\{\left[\frac{16}{(\gamma+1)\omega e^C}\right]\left[\frac{x}{\ell}\left(1 - \frac{x}{\ell}\right)\right]\right\} + \frac{\omega^2}{2}\left(\frac{1 - 3\frac{x}{\ell}}{1 - \frac{x}{\ell}}\right) - \omega^2 \tag{105a}$$

for $0 \leq x/\ell \leq 1/3$, and

$$(C_p)_B = -2\omega^2 \ln\frac{\omega x}{\ell} + \omega^2 \ln\left\{\omega^2 + \frac{4\left[1 - \frac{x^2}{\ell^2}\right]}{(\gamma+1)\omega^2 e^C}\right\} - \omega^2 \tag{105b}$$

for $1/3 \leq x/\ell \leq 1$.

Figure 18 shows sketches of the wing of $m = 1/2$, $\tau = 0.06$, the equivalent body of revolution, the pressure distributions on each at zero angle of attack, and the aerodynamic loading or difference in pressure $\Delta p/q_\infty = (C_p)_{W,\ell} - (C_p)_{W,u}$ on the two sides of the wing, where subscripts ℓ and u denote the lower and upper sides of the wing. For these parameters

$$(C_p)_W - (C_p)_B = -0.0364 \tag{106}$$

for zero angle of attack, and

$$\frac{\Delta p}{q_\infty} = \frac{4\alpha m^2 x}{(m^2x^2 - y^2)^{1/2}} = \frac{2\alpha}{(1 - y^2/s^2)^{1/2}} \tag{107}$$

where $s = mx$ is the semispan at a distance x from the apex. These results, shown in the plots of figure 18, are seen to be in satisfactory agreement with the experimental results of reference 41. Moreover, they serve to illustrate that the aerodynamic loading, and therefore the lift and other lateral forces and moments, on low-aspect-ratio wings and slender bodies depend entirely on $\phi_{2,\alpha}$; that these quantities are independent of free-stream Mach number; and that they are the same as indicated by linearized slender-body theory for subsonic or supersonic flow.

Although the two plots of figure 18 show that the differences in pressure between related points on the wing and equivalent body of revolution are predicted satisfactorily by the transonic equivalence rule, a similar comparison of the absolute values for the pressures would not display quite such good agreement because of the occurrence of unexpectedly large interference effects of the walls. To emphasize this point, figure 19 from reference 9 has been included to show the pressure distribution measured in the Ames 2-foot transonic wind tunnel on the circular cone-cylinder of figure 19, together with the corresponding results measured on the same model, of length 5.50 inches, in the Ames 14-foot transonic wind tunnel, and the theoretical results indicated by equation (105). The contention that the differences between the experimental results and the theory are due primarily to wind-tunnel-wall interference is supported not only by the fact that the differences are less for the tests in the larger wind tunnel, but also by a theoretical analysis carried out by the experimenter (ref. 42). Since these various results suggest that the differences in pressure are more reliable than the actual level of the pressures measured in the 2-foot transonic wind tunnel, we conclude that the evaluation of the transonic equivalence rule in figure 18 in terms of pressure differences is not only justified, but appropriate.

From the preceding analysis and applications, it is but a short step to the simple wing-body combinations of figures 20 and 21. In both examples, a flat plate triangular wing of zero thickness is combined with a slender body of revolution. In figure 20, the body is a slender cone with its apex at the nose of the wing; in figure 21, it is a truncated parabolic-arc

body of revolution attached to the wing in such a way that the wing root extends from 25 to 75 percent of the complete body length. In order to relate the results most directly to those discussed previously herein, the aspect ratio of the wing is taken to be 2, the semiapex angle of the conical body to be 7°, and the diameter-length ratio D/ℓ of the complete parabolic-arc body to be 1/12. At zero angle of attack, the wing does not affect the flow, and the pressure distribution is the same as on the surface, or at the corresponding point in the flow field, of the isolated body as illustrated in the left parts of these figures. The pressure distribution at zero angle on the body of the conical wing-body combination is thus given by equation (105), and that on the wing at a distance r from the axis and x from the apex, as measured along the axis, is given by equation (66) with $S(x) = \pi R^2 = \pi\omega^2 x^2$ and $u_B/U_\infty = -\frac{1}{2}\left[(C_p)_B + \omega^2\right]$. No comparable analytic expressions can be cited for the pressure distribution on the second wing-body combination at zero angle of attack, illustrated in the left part of figure 21, but the results are the same as indicated in figure 4(c) for the corresponding points on the body surface and in the surrounding flow.

The aerodynamic loadings shown in the right parts of figures 20 and 21 have been computed using equation (21) for C_p, equation (35) for ϕ, equation (44) for $\phi_{2,t}$, and the following expression from references 31 and 42 for $\phi_{2,\alpha}$:

$$\phi_{2,\alpha} = \pm \frac{U_\infty \alpha}{\sqrt{2}} \left\{ \left[-\left(1 + \frac{a_1^4}{r^4}\right) r^2 \cos 2\theta + s^2 \left(1 + \frac{a_1^4}{s^4}\right) \right] \right.$$

$$+ \left[r^4 \left(1 - \frac{a_1^4}{r^4}\right)^2 + 4a_1^4 \cos^2 2\theta + s^4 \left(1 + \frac{a_1^4}{s^4}\right)^2 \right.$$

$$\left. \left. - 2s^2 \left(1 + \frac{a_1^4}{r^4}\right)\left(1 + \frac{a_1^4}{s^4}\right) r^2 \cos 2\theta \right]^{1/2} \right\}^{1/2} - U_\infty \alpha z \qquad (108)$$

in which the sign is positive for the upper half-plane $0 \leq \theta \leq \pi$ and negative for the lower half-plane $\pi \leq \theta \leq 2\pi$, and where $a_1 = a_1(x)$ and $s = s(x)$ represent the local radius of the body and the semispan of the wing. For the specific examples described above

$$a_1 = \omega x = 0.1225\ x, \qquad s = mx = x/2 \tag{109}$$

for the conical wing-body combination; and

$$\left.\begin{aligned} a_1 &= 2D\left(\frac{x}{\ell} - \frac{x^2}{\ell^2}\right) = \frac{1}{6}\left(\frac{x}{\ell} - \frac{x^2}{\ell^2}\right) \\ s &= \frac{\ell}{2}\left(\frac{x}{\ell} - \frac{1}{4} + \frac{3D}{4\ell}\right) = \frac{\ell}{2}\left(\frac{x}{\ell} - \frac{3}{16}\right) \end{aligned}\right\} \tag{110}$$

for the wing-body combination with the parabolic-arc body. Upon carrying out the indicated operations, the following expressions are found for the aerodynamic loading

$$\frac{\Delta p}{q_\infty} = \frac{4\alpha\left\{\left[\frac{ds}{dx}\left(1 - \frac{a_1^4}{s^4}\right)\right] + \frac{a_1}{s}\,\frac{da_1}{dx}\left[2\left(\frac{a_1^2}{s^2} - 1\right) + \left(1 - \frac{a_1^2}{y^2}\right)^2\right]\right\}}{\left[1 + \frac{a_1^4}{s^4} - \frac{y^2}{s^2}\left(1 + \frac{a_1^4}{y^4}\right)\right]^{1/2}} \tag{111a}$$

on the wing, and

$$\frac{\Delta p}{q_\infty} = \frac{4\alpha\left[\frac{ds}{dx}\left(1 - \frac{a_1^4}{s^4}\right) + 2\,\frac{a_1}{s}\,\frac{da_1}{dx}\left(\frac{a_1^2}{s^2} + 1 - 2\,\frac{y^2}{a_1^2}\right)\right]}{\left[\left(1 + \frac{a_1^2}{s^2}\right)^2 - 4\,\frac{y^2}{s^2}\right]^{1/2}} \tag{111b}$$

on the body. Although we are unaware of any experimental data with which these or related results may be compared, the generally good agreement displayed in the preceding comparisons of pressure distributions on the bodies and wings separately suggest that correspondingly good agreement is to be anticipated for the wing-body combinations.

CONCLUDING REMARKS

Analysis and associated development of computational programs have been conducted in order to develop calculative techniques for predicting properties of transonic flows about certain classes of airfoils and slender bodies. The theoretical analysis, which is based on the local linearization method in the case of two-dimensional flow, combines that method with the transonic equivalence rule for studying the primary case of interest, i.e., three-dimensional transonic flows.

For the two-dimensional case, programs have been developed for nonlifting flows about symmetric airfoils with ordinates described by $Z \sim (x - x^n)$ or $(1 - x - (1 - x)^n)$ for the

- Calculation of surface pressure distributions for $M_\infty \approx 1$, $M_\infty < M_{cr,\ell}$, and $M_{cr,u} < M_\infty$
- Calculation of $M_{cr,\ell}$ and $M_{cr,u}$

Results of calculations from these programs demonstrate the remarkably wide extent of the transonic range for these airfoils, and comparisons of theoretical and experimental results at $M_\infty = 1$ serve to complete comparisons for an entire series of airfoils investigated experimentally and to reaffirm conclusions regarding the notable accuracy of the local linearization method. The differences in the comparisons that are apparent near the trailing edge were examined to determine whether they might be attributable to a purely inviscid effect as would be the case if the local Mach number immediately upstream of the trailing edge were not high enough for an oblique shock to turn the flow through the required angle so that detachment and movement of the shock onto the airfoil surface would occur. For all the airfoils tested, the local Mach number was sufficient indicating that differences between theoretical and experimental pressure distributions over the aft portion of airfoils with small or moderate trailing-edge angles must involve viscous processes.

For three-dimensional axisymmetric transonic flows about bodies of revolution with radial coordinates described by $R \sim (x - x^n)$ or $(1 - x - (1 - x)^n)$, programs have been developed for

- Calculation of surface and flow-field pressure distributions for $M_\infty \approx 1$, $M_\infty < M_{cr,\ell}$, and $M_{cr,u} < M_\infty$
- Calculation of $M_{cr,\ell}$ and $M_{cr,u}$

and for three-dimensional nonaxisymmetric transonic flows about parabolic-arc bodies having elliptical cross sections, for

- Calculation of surface and flow-field pressure distributions for $M_\infty \approx 1$, $M_\infty < M_{cr,\ell}$, and $M_{cr,u} < M_\infty$ for nonlifting flows
- Calculation of surface pressure distributions for $M_\infty \approx 1$, $M_\infty < M_{cr,\ell}$, and $M_{cr,u} < M_\infty$ for lifting flows

Altogether, comparisons of results predicted by the above programs and data obtained from wind-tunnel tests demonstrate good general agreement, indicating that the local linearization method is capable of providing accurate solutions for flow properties on the surface of a large class of axisymmetric shapes, that the transonic equivalence rule is able to account satisfactorily for moderate nonaxisymmetric effects caused by body shape and lift, and also that it can successfully extend results for flow properties calculated on the body surface out to moderate distances in the flow field.

For the axisymmetric bodies studied, the predicted surface and flow-field pressure distributions are in essential agreement over the forebodies of all the bodies tested both with measurements taken in the Ames 14-foot transonic wind tunnel and the Ames 12-foot solid-wall wind tunnel operating under choked conditions. Near the rear of these bodies, however, all three sets of results diverge widely, with the data from the choked tunnel generally falling on the opposite side of the theoretical curve than the data from the transonic wind tunnel. These facts, combined with the characteristic diagrams for these bodies, indicate that in addition to the expected shock-wave boundary-layer interaction effects occurring on the aft sections of these bodies, measurements taken on the bodies considered herein in those wind tunnels are subject to substantial wall interference effects. Results for the extent of the transonic regime for these bodies indicate a much smaller range than the corresponding results for the two-dimensional cases, emphasizing the fact that an infinite wing creates a much larger flow disturbance than a body of revolution of the same thickness ratio. Moreover, the flow-field comparisons indicate that, because of the strain imposed on the slender-body approximation by the blunter forebody of the bodies having their point of maximum thickness close to the nose, applications of the theory as it is presently constituted should be made with caution to those particular shapes.

Results for the nonlifting surface and flow-field pressure distributions on the parabolic-arc bodies with elliptic cross section indicate the remarkably small effect of the ellipticity of the cross section and the rapid tendency of the flow field to become axisymmetric. The lifting surface pressure distributions for those same bodies at angles of attack up to 6^{o} suggest that, in addition to the effects discussed previously with regard to discrepancies near the rear of the bodies, these experimental results are also being influenced by vortex separation over part of the lee surface due to the crossflow velocity component.

To demonstrate how the present results can readily be extended to more complex configurations, surface pressure distributions and loadings have been calculated for a conical wing-body combination having a flat-plate triangular wing and a more general configuration consisting of a truncated parabolic-arc body and a flat-plate triangular wing.

In conclusion, we emphasize that the procedures by which these results have been obtained are not restricted to the particular examples selected for display in this report, but possess much greater generality. Moreover, the quality of the agreement with experimental results is sufficiently good, particularly when consideration is given to the numerous shortcomings inherent in transonic wind-tunnel testing, that it is possible to go forward to more complicated configurations with confidence that the analysis is not only capable of being carried out, but that the results will be of sufficient accuracy to be useful to aerodynamicists. We suggest, furthermore, that parallel experimental work be conducted to determine surface and flow-field pressure distributions on selected wing-body combinations so that the extent to which the theory can be applied to configurations of this nature can be more clearly defined.

Nielsen Engineering & Research, Inc.
Mountain View, California
July 22, 1970

LISTING OF TWO-DIMENSIONAL COMPUTER PROGRAMS

```
C     PROGRAM FOR DETERMINING THE SUBSONIC PRESSURE DISTRIBUTION FOR
C     FREE STREAM MACH NUMBERS LESS THAN THE LOWER CRITICAL ON
C     NONLIFTING SYMMETRIC AIRFOILS HAVING ORDINATES Z PROPORTIONAL TO
C     X-X**N OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION
C     - FOR REFERENCE SEE SPREITER, J.R., AND ALKSNE, A. Y., NASA TR 1359
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--LOCATION AS FRACTION OF CHORD (X/C) OF POSITION
C     OF MAXIMUM THICKNESS OF AIRFOIL
C     COLUMNS 11 TO 20--FINENESS RATIO OF AIRFOIL
C     COLUMNS 21 TO 30--RATIO OF SPECIFIC HEATS OF GAS
C     COLUMNS 31 TO 40--FREE STREAM MACH NUMBER
C     COLUMNS 41 TO 50--INTERVAL SIZE AS FRACTION OF CHORD FOR PRESSURE
C     DISTRIBUTION PRINT-OUT
C*****************************************************************
C
C
      REAL M,MSQ,N,N1,NEXP,NUM,NEWN
      COMMON XMT,N
      DIMENSION XMTKS(19),NEXP(19)
      XMTKS(1)=.05
      DO 100 I=2,19
  100 XMTKS(I)=XMTKS(I-1)+.05
      NEXP(1)=88.731
      NEXP(2)=34.649
      NEXP(3)=19.173
      NEXP(4)=12.215
      NEXP(5)=8.396
      NEXP(6)=6.044
      NEXP(7)=4.482
      NEXP(8)=3.389
      NEXP(9)=2.595
      NEXP(10)=2.000
      DO 101 I=1,9
      J=10-I
  101 NEXP(I+10)=NEXP(J)
    1 READ (5,2) XMT,F,GAMMA,M,DX
    2 FORMAT (5E10.4)
      WRITE (6,200)
  200 FORMAT (1H1,39X64HCALCULATION OF SURFACE PRESSURE DISTRIBUTION FOR
     1 PURELY SUBSONIC / 40X59HFLOW OVER A NONLIFTING SYMMETRIC AIRFOIL
     2HAVING ORDINATES Z / 40X81HPROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
     3BY USING THE METHOD OF LOCAL LINEARIZATION ///)
  201 FORMAT (1H ,5X,1HX,12X,5HCPBAR,14X,2HCP//)
      IF (XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF (F.LT.0.) GO TO 304
      IF (GAMMA.LT.1..OR.GAMMA.GT.1.667) GO TO 305
      IF (M.GT.1..OR.M.LE.0.) GO TO 306
      IF (DX.LE.0..OR.DX.GT.1.) GO TO 307
      DO 3 I=1,19
      J=I-1
      IF (XMTKS(I)-XMT) 3,4,51
    3 CONTINUE
    4 N=NEXP(I)
      GO TO 50
   51 IF (I.EQ.1) GO TO 4
      N=NEXP(J)+(NEXP(I)-NEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
   50 CONTINUE
      A=XMT
      IF (XMT.LT..5) A=1.-XMT
      NIT=0
    5 N1=N-1.
      NIT=NIT+1
      IF (NIT.GT.20) GO TO 6
      AN1=A**N1
      NUM=1.-N*AN1
      DENOM=-AN1*(N*ALOG(A)+1.)
      NEWN=N-(NUM/DENOM)
      ERR=ABS(N/NEWN-1.)
      N=NEWN
      IF (ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (5X20HEXECUTION TERMINATED/5X87HTHE VALUE OF THE EXPONENT N
     1 CANNOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 1
    8 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) N
      WRITE (6,605) GAMMA
      WRITE (6,606) M
      WRITE (6,201)
      A=(N**(N/(N-1.)))/((N-1.)*3.1415927)
      MSQ=M*M
      FAC=MSQ*(GAMMA+1.)/F
      SIMFAC=(1./(F*F*MSQ*(GAMMA+1.)))**(1./3.)
      XI=(MSQ-1.)/(FAC**(2./3.))
      XI32=(-XI)**1.5
      X=DX
    9 A1=FCT(X)
      IF (ABS(A1+100.).LE.1.E-5) GO TO 1
      CPI=-A*A1
      FAC1=XI32+.75*CPI
      IF (FAC1.LE.0.) GO TO 10
      FAC1=FAC1**(2./3.)
      CPBAR=2.*(XI+FAC1)
      CP=SIMFAC*CPBAR
      WRITE (6,301) X,CPBAR,CP
  301 FORMAT (1H ,E10.4,2(5X,1PE12.5))
      X=X+DX
      IF((1.-X).GT..01) GO TO 9
      GO TO 1
   10 WRITE (6,302)
  302 FORMAT (114H PROGRAM HAS TERMINATED BECAUSE SPECIFIED FREE STREAM
     1MACH NUMBER M IS GREATER THAN THE LOWER CRITICAL MACH NUMBER)
      GO TO 1
  303 WRITE (6,400)
  400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
      GO TO 1
  304 WRITE (6,401)
  401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
      GO TO 1
  305 WRITE (6,402)
  402 FORMAT (/65H RATIO OF SPECIFIC HEATS MUST BE GREATER THAN 1 AND LE
     1SS THAN 5/3 )
      GO TO 1
  306 WRITE (6,403)
  403 FORMAT (113H FREE STREAM MACH NUMBER MUST BE GREATER THAN 0 AND LE
     1SS THAN THE LOWER CRITICAL MACH NUMBER WHICH IS LESS THAN 1 )
      GO TO 1
  307 WRITE (6,404)
  404 FORMAT (/87H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINTOUT MUST
     1BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  601 FORMAT (1H ,5X,38HAIRFOIL AND FLOW FIELD CHARACTERISTICS //)
  602 FORMAT (1H ,32HAIRFOIL MAX. THICKNESS AT X/C = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,16X,E12.5)
  604 FORMAT (1H ,35HEXPONENT N FOR AIRFOIL ORDINATES = ,E12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = , 9X,E12.5)
  606 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,7X,E12.5//)
      END
```

```
      FUNCTION FCT(X)
      REAL N
      EXTERNAL FUN
      COMMON XMT,N
      COMMON /BLK1/ Z,Z1
      Z=X
      IF (XMT.GE..5) GO TO 1
      Z1=(1.-X)**(N-1.)
      GO TO 2
    1 Z1=X**(N-1.)
    2 XL=0.
      XU=1.
      NIT=10
      TOL=1.E-4
      CALL SIMP (XL,XU,NIT,NIT1,TOL,FUN,ANS)
      IF (NIT1.EQ.0) GO TO 5
      IF (XMT.GE..5) GO TO 3
      DERIV=-1.+N*((1.-X)**(N-1.))
      GO TO 4
    3 DERIV=1.-N*(X**(N-1.))
    4 FCT=N*ANS-DERIV*ALOG((1.-X)/X)
      RETURN
    5 WRITE (6,6) X
    6 FORMAT (/5X28HEXECUTION TERMINATED AT X = ,E12.5/5X112HTHE INCOMPR
     1ESSIBLE PRESSURE INTEGRAL COULD NOT BE DETERMINED TO .01 PERCENT I
     2N 10 ITERATIONS USING SIMPSONS RULE)
      FCT=-100.
      RETURN
      END
```

```
      FUNCTION FUN(Y)
      REAL N,NUM
      COMMON XMT,N
      COMMON /BLK1/ Z,Z1
      IF (ABS(Z-Y).LE.1.E-5) GO TO 3
      IF (XMT.GE..5) GO TO 2
      NUM=Z1-(1.-Y)**(N-1.)
    1 FUN=NUM/(Y-Z)
      RETURN
    2 NUM=Y**(N-1.)-Z1
      GO TO 1
    3 IF (XMT.GE..5) GO TO 4
      FUN=(N-1.)*((1.-Y)**(N-2.))
      RETURN
    4 FUN=(N-1.)*(Y**(N-2.))
      RETURN
      END
```

```
      SUBROUTINE SIMP(XL,XU,NIT,NIT1,TOL,FUN,ANS)
      NIT1=10
      H=(XU-XL)/2.
      SUM1=FUN(XL)+FUN(XU)
      SUM2=FUN(XL+H)
      ANS=H*(SUM1+4.*SUM2)/3.
      N=2
      DO 3 I=1,NIT
      ANS1=ANS
      N=N*2
      H=H/2.
      SUM3=0.
      NLIM=N-1
      DO 1 K=1,NLIM,2
      AK=K
    1 SUM3=SUM3+FUN(XL+H*AK)
      ANS=H*(SUM1+2.*SUM2+4.*SUM3)/3.
      IF(ABS(ANS).LE.1.E-3) GO TO 2
      ERR=ABS(ANS1/ANS-1.)
      IF (ERR.LE.TOL) RETURN
      GO TO 3
    2 ERR=ABS(ANS-ANS1)
      IF (ERR.LT.1.E-5) RETURN
    3 SUM2=SUM2+SUM3
      NIT1=0
      RETURN
      END
```

```
C     PROGRAM FOR DETERMINING THE SUPERSONIC PRESSURE DISTRIBUTION FOR
C     FREE STREAM MACH NUMBERS ABOVE THE UPPER CRITICAL FOR
C     NONLIFTING SYMMETRIC AIRFOILS HAVING ORDINATES Z PROPORTIONAL TO
C     X-X**N OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION
C     - FOR REFERENCE SEE SPREITER, J.R. AND ALKSNE, A. Y., NASA TR 1359
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--LOCATION AS FRACTION OF CHORD (X/C) OF POSITION
C     OF MAXIMUM THICKNESS OF AIRFOIL
C     COLUMNS 11 TO 20--FINENESS RATIO OF AIRFOIL
C     COLUMNS 21 TO 30--RATIO OF SPECIFIC HEATS OF GAS
C     COLUMNS 31 TO 40--FREE STREAM MACH NUMBER
C     COLUMNS 41 TO 50--INTERVAL SIZE AS FRACTION OF CHORD FOR PRESSURE
C     DISTRIBUTION PRINT-OUT
C*********************************************************************
C
C
      REAL M,MSQ,N,N1,NEXP,NUM,NEWN
      DIMENSION XMTKS(19),NEXP(19)
      XMTKS(1)=.05
      DO 100 I=2,19
  100 XMTKS(I)=XMTKS(I-1)+.05
      NEXP(1)=88.731
      NEXP(2)=34.649
      NEXP(3)=19.173
      NEXP(4)=12.215
      NEXP(5)=8.396
      NEXP(6)=6.044
      NEXP(7)=4.482
      NEXP(8)=3.389
      NEXP(9)=2.595
      NEXP(10)=2.000
      DO 101 I=1,9
      J=10-I
  101 NEXP(I+10)=NEXP(J)
    1 READ (5,2) XMT,F,GAMMA,M,DX
    2 FORMAT (5E10.4)
      WRITE (6,200)
  200 FORMAT(1H1,39X55HCALCULATION OF SURFACE PRESSURE DISTRIBUTION FOR
     1PURELY /40X58HSUPERSONIC FLOW OVER A NONLIFTING SYMMETRIC AIRFOIL
     2HAVING /40X59HORDINATES Z PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N B
     3Y USING /40X32HTHE METHOD OF LOCAL LINEARZATION ///)
  201 FORMAT (1H ,5X,1HX,12X,5HCPBAR,14X,2HCP//)
      IF (XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF (F.LT.0.) GO TO 304
      IF (GAMMA.LT.1..OR.GAMMA.GT.1.667) GO TO 305
      IF (M.LT.1.) GO TO 306
      IF (DX.LE.0..OR.DX.GT.1.) GO TO 307
      DO 3 I=1,19
      J=I-1
      IF (XMTKS(I)-XMT) 3,4,51
    3 CONTINUE
    4 N=NEXP(I)
      GO TO 50
   51 IF (I.EQ.1) GO TO 4
      N=NEXP(J)+(NEXP(I)-NEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
   50 CONTINUE
      A=XMT
      IF (XMT.LT..5) A=1.-XMT
      NIT=0
    5 N1=N-1.
      NIT=NIT+1
      IF (NIT.GT.20) GO TO 6
      AN1=A**N1
      NUM=1.-N*AN1
      DENOM=-AN1*(N*ALOG(A)+1.)
      NEWN=N-(NUM/DENOM)
      ERR=ABS(N/NEWN-1.)
      N=NEWN
      IF (ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (5X20HEXECUTION TERMINATED/5X87HTHE VALUE OF THE EXPONENT N
     1 CANNOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 1
    8 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) N
      WRITE (6,605) GAMMA
      WRITE (6,606) M
      WRITE (6,201)
      A=(N**(N/(N-1.)))/(2.*(N-1.))
      MSQ=M**M
      FAC=MSQ*(GAMMA+1.)/F
      SIMFAC=(1./(F**MSQ*(GAMMA+1.)))**(1./3.)
      XI=(MSQ-1.)/(FAC**(2./3.))
      XI32=XI**1.5
      X=0.
    9 IF (XMT.GT..5) GO TO 10
      S=(1.-X)**(N-1.)
      SLOPE=A*(N*S-1.)
      GO TO 11
   10 S=X**(N-1.)
      SLOPE=A*(1.-N*S)
   11 FAC1=XI32-1.5*SLOPE
      IF (FAC1.LE.0.) GO TO 12
      FAC1=(FAC1)**(2./3.)
      CPBAR=2.*(XI-FAC1)
      CP=SIMFAC*CPBAR
      WRITE (6,301) X,CPBAR,CP
  301 FORMAT (1H ,E10.4,2(5X,E12.5))
      X=X+DX
      IF (X.LT.1.) GO TO 9
      GO TO 1
   12 WRITE (6,302)
  302 FORMAT(//111H PROGRAM HAS TERMINATED BECAUSE SPECIFIED FREE STREAM
     1MACH NUMBER M IS LESS THAN THE UPPER CRITICAL MACH NUMBER )
      GO TO 1
  303 WRITE (6,400)
  400 FORMAT (/43H XMT MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  304 WRITE (6,401)
  401 FORMAT (/43H FINENESS RATIO F MUST BE GREATER THAN ZERO )
      GO TO 1
  305 WRITE (6,402)
  402 FORMAT (/65H RATIO OF SPECIFIC HEATS MUST BE GREATER THAN 1 AND LE
     1SS THAN 5/3 )
      GO TO 1
  306 WRITE (6,403)
  403 FORMAT(//100H FREE STREAM MACH NUMBER MUST BE GREATER THAN THE UPPE
     1R CRITICAL MACH NUMBER WHICH IS GREATER THAN 1 )
      GO TO 1
  307 WRITE (6,404)
  404 FORMAT (/89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT MUS
     1T BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  601 FORMAT (6X   38HAIRFOIL AND FLOW FIELD CHARACTERISTICS //)
  602 FORMAT (1H ,32HAIRFOIL MAX. THICKNESS AT X/C = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,16X,E12.5)
  604 FORMAT (1H ,35HEXPONENT N FOR AIRFOIL ORDINATES = ,E12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = , 9X,E12.5)
  606 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,7X,E12.5//)
      END
```

```
C     PROGRAM FOR DETERMINING THE SURFACE PRESSURE DISTRIBUTION FOR FREE
C     STREAM MACH NUMBER M AT OR NEAR 1 FOR NONLIFTING SYMMETRIC
C     AIRFOILS HAVING ORDINATES Z PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     BY USING THE METHOD OF LOCAL LINEARIZATION - FOR REFERENCE SEE
C     SPREITER, J. R., AND ALKSNE, A. Y., NASA TR 1359
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--LOCATION AS FRACTION OF CHORD (X/C) OF POSITION
C     OF MAXIMUM THICKNESS OF AIRFOIL
C     COLUMNS 11 TO 20--FINENESS RATIO OF AIRFOIL
C     COLUMNS 21 TO 30--RATIO OF SPECIFIC HEATS OF GAS
C     COLUMNS 31 TO 40--FREE STREAM MACH NUMBER
C     COLUMNS 41 TO 50--INTERVAL SIZE AS FRACTION OF CHORD FOR PRESSURE
C     DISTRIBUTION PRINT-OUT
C*********************************************************************
C
C
      REAL INTGRL,M,MSQ,N,N1,NEWN,NEXP,NUM
      COMMON XMT,N
      DIMENSION X(1025),F(1025),XMTKS(19),NEXP(19)
      XMTKS(1)=.05
      DO 100 I=2,19
  100 XMTKS(I)=XMTKS(I-1)+.05
      NEXP(1)=88.731
      NEXP(2)=34.649
      NEXP(3)=19.173
      NEXP(4)=12.215
      NEXP(5)=8.396
      NEXP(6)=6.044
      NEXP(7)=4.482
      NEXP(8)=3.389
      NEXP(9)=2.595
      NEXP(10)=2.000
      DO 101 I=1,9
      J=10-I
  101 NEXP(I+10)=NEXP(J)
    1 READ (5,2) XMT,FIN,GAMMA,M,DX
    2 FORMAT (5E10.4)
      WRITE (6,200)
  200 FORMAT(1H1,39X60HCALCULATION OF SURFACE PRESSURE DISTRIBUTION FOR
     1FREE STREAM /40X60HMACH NUMBER M AT OR NEAR 1 FOR NONLIFTING SYMME
     2TRIC AIRFOILS /40X60HHAVING ORDINATES Z PROPORTIONAL TO X-X**N OR
     3 1-X-(1-X)**N BY /40X39HUSING THE METHOD OF LOCAL LINEARIZATION///)
      IF (XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(FIN.LE.0.) GO TO 304
      IF (GAMMA.LT.1..OR.GAMMA.GT.1.667) GO TO 305
      IF (DX.LE.0..OR.DX.GT.1.) GO TO 307
      DO 3 I=1,19
      SENTI=ABS(XMTKS(I)-XMT)
      IF(.05-SENTI) 3,4,4
    3 CONTINUE
    4 N=NEXP(I)
      A=XMT
      IF (XMT.LT..5) A=1.-XMT
      NIT=0
    5 N1=N-1.
      NIT=NIT+1
      IF (NIT.GT.20) GO TO 6
      AN1=A**N1
      NUM=1.-N*AN1
      DENOM=-AN1*(N*ALOG(A)+1.)
      NEWN=N-(NUM/DENOM)
      ERR=ABS(N/NEWN-1.)
      N=NEWN
      IF (ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (5X20HEXECUTION TERMINATED/5X87HTHE VALUE OF THE EXPONENT N
     1 CANNOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 1
    8 WRITE (6,601)
  601 FORMAT (1H ,5X38HAIRFOIL AND FLOW FIELD CHARACTERISTICS //)
      WRITE (6,602) XMT
  602 FORMAT (1H ,32HAIRFOIL MAX. THICKNESS AT X/C = ,3X,E12.5)
      WRITE (6,603) FIN
  603 FORMAT (1H ,19HFINENESS RATIO F = ,16X,E12.5)
      WRITE (6,604) N
  604 FORMAT (1H ,35HEXPONENT N FOR AIRFOIL ORDINATES = ,E12.5)
      WRITE (6,605) GAMMA
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = , 9X,E12.5)
      WRITE (6,606) M
  606 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,7X,E12.5//)
      MSQ=M*M
      Q1=(N**(N/(N-1.)))/(N-1.)
      Q2=6.*Q1*Q1/3.1415927
      SIMFAC=(1./(FIN*FIN*MSQ*(GAMMA+1.)))**(1./3.)
      X1=(MSQ*(GAMMA+1.)/FIN)**(2./3.)
      XI=(MSQ-1.)/X1
C
C     CALCULATION OF SONIC POINT
C
      CALL SONPT(XS,FS)
      IF(ABS(XS+1.)-LE.1.E-5) GO TO 1
C
C     THE FOLLOWING CARD COMPUTE THE SURFACE PRESSURE DISTRIBUTION
C
      WRITE (6,250)
  250 FORMAT (1H0,5X,1HX,12X,5HCPBAR,14X,2HCP//)
      X0=DX
      IF (XS.LT.0.1) GO TO 50
      GO TO 10
   50 IF (.09-X0) 10,40,40
   40 X0=X0+DX
      GO TO 50
   10 H=(X0-XS)/4.
      X(1)=XS
      F(1)=FS
      DO 11 I=2,5
      X(I)=X(I-1)+H
   11 F(I)=INTGRL(X(I))
      SUM1=(-3.*F(1)-10.*F(2)+18.*F(3)-6.*F(4)+F(5))**2
      SUM1=SUM1+(-F(1)+6.*F(2)-18.*F(3)+10.*F(4)+3.*F(5))**2
      SUM2=(F(1)-8.*F(2)+8.*F(4)-F(5))**2
      SUM3=(3.*F(1)-16.*F(2)+36.*F(3)-48.*F(4)+25.*F(5))**2
      CP1=(4.*SUM1+2.*SUM2+SUM3)/(432.*H)
      NI=4
      K=1
   12 J=NI+1
      H=H/2.
      NI=NI*2
      NN=NI+1
      DO 13 I=1,NN,2
      II=NN+1-I
      X(II)=X(J)
      F(II)=F(J)
   13 J=J-1
      DO 14 I=2,NI,2
      AI=I-1
      X(I)=XS+AI*H
   14 F(I)=INTGRL(X(I))
      SUM1=(-3.*F(1)-10.*F(2)+18.*F(3)-6.*F(4)+F(5))**2
      SUM1=SUM1+(-F(NI-3)+6.*F(NI-2)-18.*F(NI-1)+10.*F(NI)+3.*F(NI+1))**
     2
      JJ=NI-2
      DO 15 I=4,JJ,2
   15 SUM1=SUM1+(F(I-2)-8.*F(I-1)+8.*F(I+1)-F(I+2))**2
      MM=NI-1
      SUM2=0.
      DO 16 I=3,MM,2
   16 SUM2=SUM2+(F(I-2)-8.*F(I-1)+8.*F(I+1)-F(I+2))**2
```

```
      SUM3=(3.*F(NI-3)-16.*F(NI-2)+36.*F(NI-1)-48.*F(NI)+25.*F(NI+1))**2
      CP2=(4.*SUM1+2.*SUM2+SUM3)/(432.*H)
      IF (ABS(CP2).LE.1.E-3) GO TO 30
      ERR=ABS(CP1/CP2-1.)
      IF (ERR-.001) 20,20,17
   30 ERR=ABS(CP1-CP2)
      IF (ERR-.00001) 20,20,17
   17 IF (K-9) 18,18,19
   18 K=K+1
      CP1=CP2
      GO TO 12
   19 WRITE (6,103) X0
  103 FORMAT (/5X28HEXECUTION TERMINATED AT X = ,E12.5/5X107HTHE OUTER I
     1NTERGRAL OF THE PRESSURE COEFFICIENT CAN NOT BE DETERMINED TO WITH
     2IN .1 PERCENT IN 10 ITERATIONS )
      GO TO 1
   20 IF (X0.GT.XS) GO TO 21
      CP2=-CP2
      CP3=(Q2*CP2)**(1./3.)
      CPBAR=2.*XI+CP3
      CP=SIMFAC*CPBAR
      GO TO 22
   21 CP3=(Q2*CP2)**(1./3.)
      CPBAR=2.*XI-CP3
      CP=SIMFAC*CPBAR
   22 WRITE (6,104) X0,CPBAR,CP
  104 FORMAT (1H ,E10.4,2(5X,1PE12.5))
   23 X0=X0+DX
      IF (ABS(X0-XS).LE.1.E-3) GO TO 24
      IF (1.01-X0) 1,10,10
   24 CP2=0.
      GO TO 21
  303 WRITE (6,400)
  400 FORMAT (/43H XMT MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  304 WRITE (6,401)
  401 FORMAT (/43H FINENESS RATIO F MUST BE GREATER THAN ZERO )
      GO TO 1
  305 WRITE (6,402)
  402 FORMAT (/65H RATIO OF SPECIFIC HEATS MUST BE GREATER THAN 1 AND LE
     1SS THAN 5/3 )
      GO TO 1
  307 WRITE (6,404)
  404 FORMAT (/89H INTERNAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT MUS
     1T BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
      END



      SUBROUTINE SONPT(XS,FS)
      REAL INTGRL
      COMMON XMT,W
C     CALCULATION OF APPROXIMATE LOCATION OF SONIC POINT
      H=.05
      IF(XMT.LT..2.OR.XMT.GT..8) H=.01
      Z=H
      IF(XMT.GT..8) Z=.5
      F1=INTGRL(Z)
      Z=Z+H
      F2=INTGRL(Z)
      D0=F2
    1 Z=Z+H
      F0=F1
      F1=F2
    2 F2=INTGRL(Z)
      D1=-F0+F2
      SGN=D0*D1
      IF (SGN) 4,4,3
    3 D0=D1
      GO TO 1
C     LOCATION OF SONIC POINT IS NOW DETERMINED TO WITHIN .01 PERCENT BY
C    1USING A PARABOLIC INTERPOLATION
    4 X0=Z-H
      X2=X0-H
      X1=Z
      F2=INTGRL(X2)
      F0=INTGRL(X0)
      F1=INTGRL(X1)
      XS=X0
      XERR=10.
      N=-1
    5 N=N+1
      IF (N.GT.20) GO TO 12
      DX1=X1-X0
      DX2=X0-X2
      DXSUM=DX2+DX1
      DXDIFF=DX2-DX1
      A=DX2*F1
      B=DX1*F2
      C=DXSUM*F0
      ANUM=A*DX2-B*DX1-C*DXDIFF
      DENOM=(A+B-C)*2.
      IF(ABS(DENOM).LE.1.E-6) GO TO 499
      OLDXS=XS
      XS=X0-(ANUM/DENOM)
      FS=INTGRL(XS)
      XERR=ABS(OLDXS/XS-1.)
      IF (XERR.GT.1.E-4) GO TO 6
  500 WRITE (6,90) XS
   90 FORMAT (1H0,19HSONIC POINT AT X = ,E12.5/)
      RETURN
  499 XS=X0
      FS=INTGRL(XS)
      GO TO 500
    6 D2=ABS(X2-XS)
      D1=ABS(X1-XS)
      IF (D2.LT.D1) GO TO 9
      IF (XS.LT.X0)  GO TO 8
      IF (XS.LT.X1) GO TO 7
      X2=X0
      X0=X1
      X1=XS
      F2=F0
      F0=F1
      F1=FS
      GO TO 5
    7 X2=X0
      X0=XS
      F2=F0
      F0=FS
      GO TO 5
    8 X2=XS
      F2=FS
      GO TO 5
    9 IF (XS.GT.X0) GO TO 11
      IF (XS.GT.X2) GO TO 10
      X1=X0
      X0=X2
      X2=XS
      F1=F0
      F0=F2
      F2=FS
      GO TO 5
   10 X1=X0
      X0=XS
      F1=F0
      F0=FS
      GO TO 5
   11 X1=XS
      F1=FS
      GO TO 5
   12 WRITE (6,201)
```

```
201 FORMAT(/5X20HEXECUTION TERMINATED/5X92HTHE LOCATION OF THE SONIC P
   1OINT CAN NOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
    XS=-1.
    RETURN
    END



    FUNCTION INTGRL(X)
    REAL INTGRL,N
    EXTERNAL FUN
    COMMON XMT,N
    COMMON /BLK1/ Z,Z1
    Z=X
    IF (XMT.GE..5) GO TO 1
    Z1=(1.-X)**(N-1.)
    GO TO 2
  1 Z1=X**(N-1.)
  2 TOL=1.E-3
    NIT=10
    XL=0.
    XU=X
    CALL SIMP (XL,XU,NIT,NIT1,TOL,FUN,ANS)
    IF (NIT1.EQ.0) GO TO 4
    IF (XMT.GE..5) GO TO 3
    INTGRL=N*ANS+2.*(-1.+N*Z1)*SQRT(X)
    RETURN
  3 INTGRL=N*ANS+2.*(1.-N*Z1)*SQRT(X)
    RETURN
  4 WRITE (6,5) X
  5 FORMAT (/5X28HEXECUTION TERMINATED AT X = ,E12.5/5X75HTHE INNER IN
   1TEGRAL INVOLVED IN DETERMINING THE PRESSURE DISTRIBUTION AT M=1 ,/
   2 5X80HCANNOT BE CALCULATED TO WITHIN .01 PERCENT IN 10 ITERATIONS
   3USING SIMPSON-S RULE )
    STOP
    END



    FUNCTION FUN(Y)
    REAL N,NUM
    COMMON XMT,N
    COMMON /BLK1/ Z,Z1
    IF (ABS(Z-Y).LE.1.E-3) GO TO 3
    IF (XMT.GE..5) GO TO 2
    NUM=(1.-Y)**(N-1.)-Z1
  1 FUN=NUM/SQRT(Z-Y)
    RETURN
  2 NUM=Z1-Y**(N-1.)
    GO TO 1
  3 FUN=0.
    RETURN
    END
```

```
    SUBROUTINE SIMP(XL,XU,NIT,NIT1,TOL,FUN,ANS)
    NIT1=10
    H=(XU-XL)/2.
    SUM1=FUN(XL)+FUN(XU)
    SUM2=FUN(XL+H)
    ANS=H*(SUM1+4.*SUM2)/3.
    N=2
    DO 3 I=1,NIT
    ANS1=ANS
    N=N*2
    H=H/2.
    SUM3=0.
    NN=N-1
    DO 1 K=1,NN,2
    AK=K
  1 SUM3=SUM3+FUN(XL+AK*H)
    ANS=H*(SUM1+2.*SUM2+4.*SUM3)/3.
    IF (ABS(ANS).LE.1.E-3) GO TO 2
    ERR=ABS(ANS1/ANS-1.)
    IF (ERR.LE.TOL) RETURN
    GO TO 3
  2 ERR=ABS(ANS-ANS1)
    IF (ERR.LE.1.E-6) RETURN
  3 SUM2=SUM2+SUM3
    NIT1=0
    RETURN
    END
```

```
C     PROGRAM FOR DETERMINING THE LOWER CRITICAL MACH NUMBER FOR NONLIFT
C     ING SYMMETRIC AIRFOILS HAVING ORDINATES Z PROPORTIONAL TO X-X**N
C     OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION--FOR RE
C     FERENCE SEE SPREITER,J.R. AND ALKSNE,A.Y.,NASA TR 1359
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--LOCATION AS FRACTION OF CHORD (X/C) OF POSITION
C     OF MAXIMUM THICKNESS OF AIRFOIL
C     COLUMNS 11 TO 20--FINENESS RATIO OF AIRFOIL
C     COLUMNS 21 TO 30--RATIO OF SPECIFIC HEATS OF GAS
C*********************************************************************
C
C
      REAL M,N,N1,NEWN,NEWM,NEWY,NEXP,NUM
      COMMON XMT,N
      DIMENSION XMTKS(19),NEXP(19)
      XMTKS(1)=.05
      DO 100 I=2,19
  100 XMTKS(I)=XMTKS(I-1)+.05
      NEXP(1)=88.731
      NEXP(2)=34.649
      NEXP(3)=19.173
      NEXP(4)=12.215
      NEXP(5)=8.396
      NEXP(6)=6.044
      NEXP(7)=4.482
      NEXP(8)=3.389
      NEXP(9)=2.595
      NEXP(10)=2.000
      DO 101 I=1,9
      J=10-I
  101 NEXP(I+10)=NEXP(J)
    1 READ (5,2) XMT,F,GAMMA
    2 FORMAT (3E10.4)
      TAU=1./F
      WRITE (6,200)
  200 FORMAT(1H1,39X58HCALCULATION OF LOWER CRITICAL MACH NUMBER FOR A N
     1ONLIFTING /40X59HSYMMETRIC AIRFOIL HAVING ORDINATES Z PROPORTIONAL
     2 TO X-X**N /40X58HOR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LIN
     3EARIZATION ///)
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LE.0.) GO TO 304
      IF(GAMMA.LT.1..OR.GAMMA.GT.1.667) GO TO 305
      DO 3 I=1,19
      J=I-1
      IF (XMTKS(I)-XMT) 3,4,51
    3 CONTINUE
    4 N=NEXP(I)
      GO TO 50
   51 IF (I.EQ.1) GO TO 4
      N=NEXP(J)+(NEXP(I)-NEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
   50 CONTINUE
      A=XMT
      IF(XMT.LT..5)  A=1.-XMT
      NIT=0
    5 N1=N-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**N1
      NUM=1.-N*AN1
      DENOM=-AN1*(N*ALOG(A)+1.)
      NEWN=N-(NUM/DENOM)
      ERR=(N/NEWN-1.)
      N=NEWN
      IF(ABS(ERR).GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (5X20HEXECUTION TERMINATED/5X87HTHE VALUE OF THE EXPONENT N
     1 CANNOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 1
    8 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) N
      WRITE (6,605) GAMMA
      F0=0.
      F1=0.
      F2=0.
      X0=0.
      X1=0.
      X2=0.
      X=0.
      DDX=.05
      F=-1.F6
    9 F2=F0
      F0=F1
      F1=F
      X2=X0
      X0=X1
      X1=X
      IF(XMT.LT..2.OR.XMT.GT..8) DDX=.01
      X=X+DDX
      IF(X.GE.1.) GO TO 10
      F=FCT(X)
      IF(ABS(F+100.) .LE.1.E-5) GO TO 1
      IF(F.GT.F1) GO TO 9
      GO TO 12
   10 WRITE (6,11)
   11 FORMAT (/72H PROGRAM TERMINATED BECAUSE NO PRESSURE MINIMUM WAS FO
     1UND ON THE AIRFOIL )
      GO TO 1
   12 XERR=1.E6
      NIT=-1
      XST=X0
   13 NIT=NIT+1
      IF(NIT.GT.20) GO TO 20
      DX1=X1-X0
      DX2=X0-X2
      DXSUM=DX1+DX2
      DXDIFF=DX2-DX1
      A1=DX2*F1
      B1=DX1*F2
      C1=DXSUM*F0
      ANUM=A1*DX2-B1*DX1-C1*DXDIFF
      DENOM=(A1+B1-C1)*2.
      IF(ABS(DENOM).LE.1.F-6) GO TO 499
      OLDXST=XST
      XST=X0-(ANUM/DENOM)
      FST=FCT(XST)
      XERR=ABS(OLDXST/XST-1.)
      IF(XERR.GT.1.E-4) GO TO 14
      IF(ABS(F+100.) .LE.1.E-5) GO TO 1
      GO TO 21
  499 XST=X0
      FST=FCT(XST)
      GO TO 21
   14 D2=ABS(X2-XST)
      D1=ABS(X1-XST)
      IF(D2.LT.D1) GO TO 17
      IF(XST.LT.X0) GO TO 16
      IF(XST.LT.X1) GO TO 15
      X2=X0
      X0=X1
      X1=XST
      F2=F0
      F0=F1
      F1=FST
      GO TO 13
   15 X2=X0
      X0=XST
      F2=F0
```

```
      F0=FST
      GO TO 13
   16 X2=XST
      F2=FST
      GO TO 13
   17 IF(XST.GT.X0) GO TO 19
      IF (XST.GT.X2) GO TO 18
      X1=X0
      X0=X2
      X2=XST
      F1=F0
      F0=F2
      F2=FST
      GO TO 13
   18 X1=X0
      X0=XST
      F1=F0
      F0=FST
      GO TO 13
   19 X1=XST
      F1=FST
      GO TO 13
   20 WRITE (6,704)
  704 FORMAT (/99H PROGRAM TERMINATED BECAUSE THE PARABOLIC INTERPOLATIO
     1N SCHEME CANNOT DETERMINE THE LOCATION OF THE / 2X103HINCOMPRESSIB
     2LE PRESSURE COEFFICIENT TO WITHIN .01 PERCENT IN 20 ITERATIONS IN
     3THE VICINITY OF THE POINT )
   21 B=(N**(N/(N-1.)))/(N-1.)
      C=.75*(GAMMA+1.)*B*FST*TAU/3.1415927
      DELTY=.1
      NIT=0
   22 DELTY=DELTY/2.
      YOLD=0.
      Y=DELTY
      NIT=NIT+1
   23 Y1=1.-Y
      Y2=(C*Y)**(2./3.)
      IF(Y2-Y1) 24,24,25
   24 YOLD=Y
      Y=Y+DELTY
      IF(Y.GE.1.) GO TO 800
      GO TO 23
   25 Y=(Y+YOLD)/2.
      M=SQRT(Y)
   27 Y1=1.-Y
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 28
      SRY1=SQRT(Y1)
      NUM=C*Y-Y1*SRY1
      DENOM=C+1.5*SRY1
      NEWY=Y-(NUM/DENOM)
      IF(NEWY.LE.0.) GO TO 22
      NEWM=SQRT(NEWY)
      ERRM=ABS(M/NEWM-1.)
      Y=NEWY
      M=NEWM
      IF(ERRM.GT.1.E-4) GO TO 27
      WRITE (6,310) M
  310 FORMAT (1H ,29HLOWER CRITICAL MACH NUMBER = ,6X,E12.5)
      GO TO 1
   28 WRITE (6,311)
  311 FORMAT (/99H PROGRAM HAS TERMINATED BECAUSE THE VALUE OF THE LOWER
     1CRITICAL MACH NUMBER COULD NOT BE DETERMINED / 2X38HTO WITHIN .01
     2PERCENT IN 20 ITERATIONS )
      GO TO 1
  601 FORMAT (6X, 38HAIRFOIL AND FLOW FIELD CHARACTERISTICS //)
  602 FORMAT (1H ,32HAIRFOIL MAX. THICKNESS AT X/C = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,16X,E12.5)
  604 FORMAT (1H ,35HEXPONENT N FOR AIRFOIL ORDINATES = ,E12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,9X,E12.5)
  303 WRITE (6,400)
  400 FORMAT (/43H XMT MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  304 WRITE (6,401)
  401 FORMAT (/43H FINENESS RATIO F MUST BE GREATER THAN ZERO )
      GO TO 1
  305 WRITE (6,402)
  402 FORMAT (/65H RATIO OF SPECIFIC HEATS MUST BE GREATER THAN 1 AND LE
     1SS THAN 5/3 )
      GO TO 1
  800 WRITE (6,801)
  801 FORMAT (/62H PROGRAM TERMINATED BECAUSE THE APPROXIMATE VALUE OF M
     1CR LOWER / 2X65HNEEDED FOR STARTING THE NEWTON-RAPHSON ITERATION S
     2CHEME EXCEEDS 1 )
      GO TO 1
      END
```

```
      FUNCTION FCT(X)
      REAL N
      EXTERNAL FUN
      COMMON XMT,N
      COMMON /BLK1/ Z,Z1
      Z=X
      IF(XMT.GE..5) GO TO 1
      Z1=(1.-X)**(N-1.)
      GO TO 2
    1 Z1=X**(N-1.)
    2 XL=0.
      XU=1.
      TOL=1.E-4
      NIT=10
      CALL SIMP(XL,XU,NIT,NIT1,TOL,FUN,ANS)
      IF(NIT1.EQ.0) GO TO 5
      IF(XMT.GE..5) GO TO 3
      DERIV=-1.+N*((1.-X)**(N-1.))
      GO TO 4
    3 DERIV=1.-N*(X**(N-1.))
    4 FCT=N*ANS-DERIV*ALOG((1.-X)/X)
      RETURN
    5 WRITE (6,6) X
    6 FORMAT (/5X28HEXECUTION TERMINATED AT X = ,E12.5/5X112HTHE INCOMPR
     1ESSIBLE PRESSURE INTEGRAL COULD NOT BE DETERMINED TO .01 PERCENT I
     2N 10 ITERATIONS USING SIMPSONS RULE)
      FCT=-100.
      RETURN
      END
```

```
      FUNCTION FUN(Y)
      REAL N,NUM
      COMMON XMT,N
      COMMON /BLK1/ Z,Z1
      IF(ABS(Z-Y).LT.1.E-3) GO TO 3
      IF(XMT.GE..5) GO TO 2
      NUM=Z1-(1.-Y)**(N-1.)
    1 FUN=NUM/(Y-Z)
      RETURN
    2 NUM=Y**(N-1.)-Z1
      GO TO 1
    3 IF(XMT.GE..5) GO TO 4
      FUN=(N-1.)*((1.-Y)**(N-2.))
      RETURN
    4 FUN=(N-1.)*(Y**(N-2.))
      RETURN
      END
```

```
      SUBROUTINE SIMP(XL,XU,NIT,NIT1,TOL,FUN,ANS)
      NIT1=10
      H=(XU-XL)/2.
      SUM1=FUN(XL)+FUN(XU)
      SUM2=FUN(XL+H)
      ANS=H*(SUM1+4.*SUM2)/3.
      N=2
      DO 3 I=1,NIT
      ANS1=ANS
      N=N*2
      H=H/2.
      SUM3=0.
      NLIM=N-1
      DO 1 K=1,NLIM,2
      AK=K
    1 SUM3=SUM3+FUN(XL+H*AK)
      ANS=H*(SUM1+2.*SUM2+4.*SUM3)/3.
      IF(ABS(ANS).LE.1.E-3) GO TO 2
      ERR=ABS(ANS1/ANS-1.)
      IF(ERR.LE.TOL) RETURN
      GO TO 3
    2 ERR=ABS(ANS-ANS1)
      IF(ERR.LE.1.E-6) RETURN
    3 SUM2=SUM2+SUM3
      NIT1=0
      RETURN
      END
```

```
C     PROGRAM FOR DETERMINING THE UPPER CRITICAL MACH NUMBER FOR
C     NONLIFTING SYMMETRIC AIRFOILS HAVING ORDINATES Z PROPORTIONAL TO
C     X-X**N OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION
C     - FOR REFERENCE SEE SPREITER, J.R. AND ALKSNE, A. Y., NASA TR 1359
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--LOCATION AS FRACTION OF CHORD (X/C) OF POSITION
C     OF MAXIMUM THICKNESS OF AIRFOIL
C     COLUMNS 11 TO 20--FINENESS RATIO OF AIRFOIL
C     COLUMNS 21 TO 30--RATIO OF SPECIFIC HEATS OF GAS
C*****************************************************************
C
C
      REAL M,N,N1,NEXP,NUM,NEWN,NEWM,NEWY
      DIMENSION XMTKS(19),NEXP(19)
      XMTKS(1)=.05
      DO 100 I=2,19
  100 XMTKS(I)=XMTKS(I-1)+.05
      NEXP(1)=88.731
      NEXP(2)=34.649
      NEXP(3)=19.173
      NEXP(4)=12.215
      NEXP(5)=8.396
      NEXP(6)=6.044
      NEXP(7)=4.482
      NEXP(8)=3.389
      NEXP(9)=2.595
      NEXP(10)=2.000
      DO 101 I=1,9
      J=10-I
  101 NEXP(I+10)=NEXP(J)
    1 READ (5,2) XMT,F,GAMMA
    2 FORMAT (3E10.4)
      WRITE (6,200)
  200 FORMAT(1H1,39X69HCALCULATION OF UPPER CRITICAL MACH NUMBER FOR A N
     1ON LIFTING SYMMETRIC /40X79HAIRFOIL HAVING ORDINATES Z PROPORTIONA
     2L TO X-X**N OR 1-X-(1-X)**N  BY USING THE  /40X29HMETHOD OF LOCAL
     3LINEARIZATION ///)
      IF (XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF (F.LT.0.) GO TO 304
      IF (GAMMA.LT.1..OR.GAMMA.GT.1.667) GO TO 305
      DO 3 I= 1,19
      J=I-1
      IF (XMTKS(I)-XMT) 3,4,51
    3 CONTINUE
    4 N=NEXP(I)
      GO TO 50
   51 IF (I.EQ.1) GO TO 4
      N=NEXP(J)+(NEXP(I)-NEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
   50 CONTINUE
      A=XMT
      IF (XMT.LT..5) A=1.-XMT
      NIT=0
    5 N1=N-1.
      NIT=NIT+1
      IF (NIT.GT.20) GO TO 6
      AN1=A**N1
      NUM=1.-N*AN1
      DENOM=-AN1*(N*ALOG(A)+1.)
      NEWN=N-(NUM/DENOM)
      ERR=ABS(N/NEWN-1.)
      N=NEWN
      IF (ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (5X20HEXECUTION TERMINATED/5X87HTHE VALUE OF THE EXPONENT N
     1 CANNOT BE DETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 1
    8 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) N
      WRITE (6,605) GAMMA
      S=.5*(N**(N/(N-1.)))
      IF (XMT.GT..5) S=S/(N-1.)
      C=1.5*(GAMMA+1.)*S/F
      DELTY=.1
      NIT=0
    9 DELTY=DELTY/2.
      YOLD=1.
      Y=1.+DELTY
   10 Y1=Y-1.
      Y2=(C*Y)**(2./3.)
      IF (Y1-Y2) 11,11,12
   11 YOLD=Y
      Y=Y+DELTY
      GO TO 10
   12 Y=(Y+YOLD)/2.
      M=SQRT(Y)
   13 Y1=Y-1.
      NIT=NIT+1
      IF (NIT.GT.20) GO TO 14
      SRY1=SQRT(Y1)
      NUM=C*Y-Y1*SRY1
      DENOM=C-1.5*SRY1
      NEWY=Y-(NUM/DENOM)
      IF (NEWY.LT.0.) GO TO 9
      NEWM=SQRT(NEWY)
      ERRM=ABS(M/NEWM-1.)
      Y=NEWY
      M=NEWM
      IF (ERRM.GT.1.E-4) GO TO 13
      WRITE (6,606) M
      GO TO 1
   14 WRITE (6,607)
  607 FORMAT (/39X, 62HPROGRAM HAS TERMINATED BECAUSE THE VALUE OF THE U
     1PPER CRITICAL /40X74HMACH NUMBER COULD NOT BE DETERMINED TO WITHIN
     2 .01 PERCENT IN 20 ITERATIONS )
      GO TO 1
  303 WRITE (6,400)
  400 FORMAT (/43H XMT MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 1
  304 WRITE (6,401)
  401 FORMAT (/43H FINENESS RATIO F MUST BE GREATER THAN ZERO )
      GO TO 1
  305 WRITE (6,402)
  402 FORMAT (/65H RATIO OF SPECIFIC HEATS MUST BE GREATER THAN 1 AND LE
     1SS THAN 5/3 )
      GO TO 1
  601 FORMAT(1H ,5X38HAIRFOIL AND FLOW FIELD CHARACTERISTICS //)
  602 FORMAT (1H ,32HAIRFOIL MAX. THICKNESS AT X/C = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,16X,E12.5)
  604 FORMAT (1H ,35HEXPONENT N FOR AIRFOIL ORDINATES = ,E12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = , 9X,E12.5)
  606 FORMAT (1H ,29HUPPER CRITICAL MACH NUMBER = ,6X,E12.5)
      END
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESSURE DISTRI
C     BUTION FOR FREE STREAM MACH NUMBER M BELOW THE LOWER CRITICAL ON
C     NONLIFTING BODIES OF REVOLUTION HAVING ORDINATES R PROPORTIONAL TO
C     X-X**N OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION
C     FOR REFERENCE SEE SPREITER,J.R. AND ALKSNE,A.Y.,NASA TR-R2
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--LOCATION AS FRACTION OF BODY LENGTH (X/L) OF
C     POSITION OF MAXIMUM BODY THICKNESS
C     COLUMNS 21 TO 30--FINENESS RATIO OF BODY
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 50--INDEX FOR FLOW FIELD PRINT-OUT***INDEX EQUAL TO
C     OR LESS THAN 0.0 FOR NO FLOW FIELD CALCULATION AND GREATER THAN
C     0.0 FOR FLOW FIELD CALCULATION
C*****************************************************************
C
C
C     MAIN PROGRAM
      REAL M
      DIMENSION DPRMT(5),Y(1),DDERY(1),DAUX(16,1)
      DIMENSION XMTKS(19),DNEXP(19),DZST(19)
      EXTERNAL FCT,OUTP
      COMMON /A/ DM,DK
      COMMON /B/ MM
      COMMON /C/ DCTAU2,DC
      COMMON /L/ K
      COMMON /P/ DC
      COMMON /MRMC/ DWRITE,DMRM,FLWFLD
      DATA XMTKS/0.05,.10,.15,.20,.25,.30,.35,.40,.45,.50,.55,.60,.65,
     1           .70,.75,.80,.85,.90,.95/
      DATA DNEXP/88.731,34.649,19.173,12.215,8.396,5.0[illegible],3.389,
     1           2.595,2.0[illegible],1.565,1.215,
     2           34.649,88.731/
      DATA DZST/0.0077126,0.019182,0.033578,0.050724,0.070564,0.093097,
     1          0.118356,0.146406,0.177347,0.211325,0.247310,0.285355,
     2          0.406294,0.479059,0.555659,0.636159,0.720673,0.809353,
     3          0.9023867
   10 READ (5,200) M,XMT,F,DX,FLWFLD
  200 FORMAT (5F10.5)
      WRITE(6,600)
  600 FORMAT(1H1,35X61HCALCULATION OF SURFACE AND FLOW FIELD PRESSURE DI
     1STRIBUTIONS / 36X62HFOR PURELY SUBSONIC FLOW ABOUT A NONLIFTING BO
     2DY OF REVOLUTION / 35X57HHAVING ORDINATES R PROPORTIONAL TO X-X**N
     3 OR 1-X-(1-X)**N /36X42HBY USING THE METHOD OF LOCAL LINEARIZATION
     4///)
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LT.0.) GO TO 304
      IF(M.GT.1..OR.M.LE.0.) GO TO 306
      IF(DX.LE.0..OR.DX.GT.1.) GO TO 307
      DMM=DX
      DM=M
      K=1
      IF(XMT.GT..5) K=0
      DO 913 I=1,19
      J=I-1
      IF(XMTKS(I)-XMT) 913,914,951
  913 CONTINUE
  914 DN=DNEXP(I)
      DZS1=DZST(I)
      GO TO 950
  951 IF (I.EQ.1) GO TO 914
      DN=DNEXP(J)+(DNEXP(I)-DNEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
      DZS1=DZST(J)+(DZST(I)-DZST(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
  950 CONTINUE
      A=XMT
      IF(XMT.LT..5) A=1.-XMT
      NIT=0
    5 DN1=DN-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**DN1
      DNUM=1.-DN*AN1
      DENOM=-AN1*(DN*ALOG(A)+1.)
      DNEWN=DN-(DNUM/DENOM)
      ERR=ABS(DN/DNEWN-1.)
      DN=DNEWN
      IF(ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (4X49HEXECUTION TERMINATED BECAUSE EXPONENT N CANNOT BE/5X
     149HDETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 10
    8 DC=(DN**(DN/(DN-1.00)))/(DN-1.00)
      MM=1
      DTAU=1./F
      GAMMA=1.4
      DCTAU=DC*DTAU
      DCTAU2=DCTAU*DCTAU
      DZJ=0.01
      DZF=0.99
      DIM=1.00-DM*DM
      DK=DM*DM*2.4
C
C     CALCULATION OF THE POINT WHERE S--(X) = 0
C
      N=1
    1 DZS=DZS1-DS2PI(DZS1)/DS3PI(DZS1)
      IF(ABS(DZS-DZS1).LT.1.E-6) GO TO 3
      DZS1=DZS
      N=N+1
      IF(N.GT.10) GO TO 2
      GO TO 1
    2 WRITE (6,100)
  100 FORMAT (1H0,4X55HEXECUTION TERMINATED BECAUSE S--(X) = 0 POINT CAN
     1NOT BE /5X57HDETERMINED TO WITHIN SUFFICIENT ACCURACY IN 10 ITERAT
     2IONS)
      GO TO 10
C
C     START OF INTEGRATION PROCEDURE
C
    3 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) DN
      WRITE (6,605) DZS
      WRITE (6,606) GAMMA
      WRITE (6,607) M
      DMRM=0.
      IF (DMM-.001) 920,920,921
  921 DWRITE=DZS/DMM  + 1.0
      MRM=DWRITE
      DWRITE=MRM
      DWRITE=DWRITE*DMM + 0.001
      DMRM=-DMM
  920 CONTINUE
      DU0=DINT(DZS)
      DRS0=-2.00*DU0-DRSQ(DZS)
      WRITE (6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO NOSE//)
      IF (FLWFLD) 30,30,31
   30 WRITE (6,102)
  102 FORMAT (5X,1HX,7X8HCP(BODY)//)
      GO TO 32
   31 WRITE (6,112)
  112 FORMAT (5X,1HX,7X8HCP(BODY),5X6HCP(1D),6X6HCP(2D),6X,6HCP(3D),
     1       6X6HCP(4D),6X6HCP(5D),6X6HCP(6D),6X6HCP(7D),6X6HCP(8D)//)
   32 CONTINUE
      DY(1)=0.0
      DDERY(1)=1.0
```

```
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
   24 DPRMT(3)=-1.0E-3
      DPRMT(4)=1.E-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      IF (DMRM.EQ.0.0) GO TO 922
      MRM=MRM-1
      DWRTF=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 DY(1)=0.0
      DDERY(1)=1.0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=1.0E-3
      DPRMT(4)=1.E-6
      NDIM=1
      WRITE (6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO TAIL//)
      IF (FLWFLD) 40,40,41
   40 WRITE (6,102)
      GO TO 42
   41 WRITE (6,112)
   42 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      GO TO 10
  601 FORMAT (1H ,49HBODY OF REVOLUTION AND FLOW FIELD CHARACTERISTICS//
     1)
  602 FORMAT (1H ,29HBODY MAX. THICKNESS AT X/L = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,13X,E12.5)
  604 FORMAT (1H ,32HEXPONENT N FOR BODY ORDINATES = ,E12.5)
  605 FORMAT (1H ,20HS--(X) = 0 AT X/L = ,12X,E12.5)
  606 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,6X,E12.5)
  607 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,4X,E12.5)
  303 WRITE (6,400)
  400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
      GO TO 10
  304 WRITE (6,401)
  401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
      GO TO 10
  306 WRITE (6,402)
  402 FORMAT (113H FREE STREAM MACH NUMBER MUST BE GREATER THAN 0 AND LE
     1SS THAN THE LOWER CRITICAL MACH NUMBER WHICH IS LESS THAN 1 )
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (/89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT MUS
     1T BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END
```

```
      SUBROUTINE FCT(DX,DY,DZ)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /F/ DEF
      DA=DSPI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      DEF=DA2*ALOG(DA/(DX*(1.00-DX)))+DINT(DX)
      DUU=D1M-DK*(DY(1)+DEF)
      IF(DUU) 2,2,1
    1 DZ(1)=DA3*ALOG(DUU)
      RETURN
    2 MM=2
      WRITE(6,700) DX
  700 FORMAT (1H0,42HLOG ARGUMENT (1-MM-KU) IS NEGATIVE AT X = ,E12.7)
      WRITE (6,701)
  701 FORMAT (1H ,72HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER GREATE
     1R THAN LOWER CRITICAL)
      RETURN
      END
```

```
      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
      DIMENSION DY(1),DDERY(1),DPRMT(5)
      DIMENSION DCPF(8)
      COMMON /F/ DHF
      COMMON /MRMC/ DWRTE,DMRM,FLWFLD
      DU=DY(1)+DHF
      DCP=-2.00*DU-DRSQ(DX)
      IF (IHLF.GT.10) GO TO 10
      IF (DX.LE.0.01 .OR. DX.GE.0.99) GO TO 20
      IF (DMRM) 50,20,60
   50 IF (DX.GT.DWRTF) RETURN
      GO TO 20
   60 IF (DX.LT.DWRTF) RETURN
   20 IF (FLWFLD) 30,30,31
   30 WRITE (6,101) DX,DCP
      GO TO 32
   31 D1=-2.0*DU
      D2=-DRSQ(DX)
      D3=-4.*DS2PI(DX)
      D4=DRRB(DX)
      DO 7 I=1,8
      DA=1
      DB=DA*D4
    7 DCPF(I)=D1+D2/(DB*DB)+D3*ALOG(DB)
      WRITE (6,101) DX,DCP,DCPF
  101 FORMAT (1X,F8.5,1X1PE11.4,1X,8(1X1PE11.4))
   32 CONTINUE
      DWRTF=DWRTF+DMRM
      RETURN
   10 WRITE (6,100)
  100 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      GO TO 20
      END
```

```
      FUNCTION DINT(DZ)
      EXTERNAL DFUN,DFUN1
      COMMON /G/ DW,DA2
      DW=DZ
      DA2=DS2PI(DZ)
      NIT=10
      DTOL=1.E-6
      CALL SIMP(0.00,DZ,NIT,NIT1,DTOL,DFUN,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT1=DANS
      CALL SIMP(DZ,1.00,NIT,NIT1,DTOL,DFUN1,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT2=DANS
      DINT=DINT1+DINT2
      RETURN
   20 WRITE(6,21)
   21 FORMAT (1HO,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
     1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
     2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X,  40HFOR THE GIVEN NUMBE
     3R OF ITERATIONS (NIT))
      WRITE(6,22) DZ
   22 FORMAT(1H ,5X,4HX = ,E14.7)
      WRITE (6,23) DTOL
   23 FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,E14.7)
      STOP
      END
```

```
      FUNCTION DFUN(DZ)
      COMMON /G/ DW,DA2
      IF (ABS(DZ-DW).LT.1.E-6) GO TO 20
      DB2=DS2PI(DZ)
      DFUN=(DA2-DB2)/(DW-DZ)
      RETURN
   20 DFUN=DS3PI(DW)
      RETURN
      END
```

```
      FUNCTION DFUN1(DZ)
      COMMON /G/ DW,DA2
      IF (ABS(DZ-DW ).LT.1.E-6) GO TO 20
      DB2=DS2PI(DZ)
      DFUN1=(DA2-DB2)/(DZ-DW )
      RETURN
   20 DFUN1=-DS3PI(DW)
      RETURN
      END
```

```
      FUNCTION DSPI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**DN
    1 DSPI=DCTAU2*DS1*DS1/16.0
      RETURN
    2 DS1=1.00-DZ-(1.00-DZ)**DN
      GO TO 1
      END
```

```
      FUNCTION DS2PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
    1 DS2PI=DCTAU2*(1.00-DN*DS1*(DN+1.00-(2.00*DN-1.00)*DS1))/8.0
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      GO TO 1
      END
```

```
      FUNCTION DS3PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(-DN-1.00+2.00*(2.00*DN-1.00)*DS1
     1*DZ))/8.0
      RETURN
    2 DS1=(1.00-DZ)**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(DN+1.00-2.00*(2.00*DN-1.00)*DS1*
     1((1.00-DZ)))/8.0
      RETURN
      END
```

```
      FUNCTION DRSQ(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
      DS2=1.00-DN*DS1
    1 DRSQ=DCTAU2*DS2*DS2/4.0
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      DS2=-1.00+DN*DS1
      GO TO 1
      END
```

```
      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      NIT1=10
      DH=(DXU-DXL)/2.0
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.00*DSUM2)/3.0
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.0
      DSUM3=0.0
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DK=K
    2 DSUM3=DSUM3+FUN(DXL+DH*DK)
      DANS=DH*(DSUM1+2.00*DSUM2+4.00*DSUM3)/3.0
      DERR=(DANS-DANS1)
      IF (ABS(DERR).LE.DTOL) RETURN
    1 DSUM2=DSUM2+DSUM3
      DTOL=DERR
      NIT1=0
      RETURN
      END
```

```
      FUNCTION DRRR(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      COMMON /P/ DC
      IF (K.GT.0) GO TO 2
      DA=DC*(DC-DZ**DN)
    1 DRRR=2./DA
      RETURN
    2 DA=DC*(1.-DZ-(1.-DZ)**DN)
      GO TO 1
      END



      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
C
      COMMON /B/ MM
      DIMENSION PRMT(1),Y(1),DERY(1),AUX(16,1)
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.0
      DO 1 I=1,NDIM
      AUX(16,I)=0.0
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C     ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C     COMPUTATION OF DERY FOR STARTING VALUES
    4 CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
C
C     RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C     COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.5*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(9,I)=DERY(I)
      ISW=3
      GO TO 100
C
C     COMPUTATION OF TEST VALUE DELT
   15 DELT=0.0
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.066666667*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C     NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.
   18 IHLF=11
      X=X+H
      GO TO 4
C
C     THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
   22 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.79166667*AUX(9,I)-.20833333*AUX(10
     1,I)+.04166667*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))16,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+.333333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+.375  *H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     ********************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+.4*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
```

```
      Z=X+.4*Z
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+.29697761*AUX(5,I)+.15875964*Z
C
      Z=X+.45573725*H
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+.21810039*AUX(5,I)-3.05095515*AUX(6,I)+3.83286476*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 104 I=1,NDIM
  104 Y(I)=AUX(N,I)+.17476028*AUX(5,I)-.55148066*AUX(6,I)+1.20553560*
     1AUX(7,I)+.17118478*H*DERY(I)
      GO TO(9,13,15,21),ISW
C     ****************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
C
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
      DELT=AUX(N-4,I)+1.33333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+AUX
     1(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-.92561983*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
      DELT=.125  *(9.00*AUX(N-1,I)-AUX(N-3,I)+3.00*H*(DERY(I)+AUX(N+6,I)
     1 +AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+.07438016*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0.0
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF(MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF (ABS(X-PRMT(2))-.1  *ABS(H))212,215,215
  215 IF(DELT-.02  *PRMT(4))216,216,201
C
C
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  221 AUX(16,I)=9.96296296*(Y(I)-AUX(N-3,I))-3.36111111*H*(DERY(I)+DELT
     1+AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=.5*H
      ISTEP=0
      DO 224 I=1,NDIM
      Y(I)=.390625E-2*(8.E1*AUX(N-1,I)+135.00*AUX(N-2,I)+4.E1*AUX(N-3,I)
     1+AUX(N-4,I))-.1171875  *(AUX(N+6,I)-6.00*AUX(N+5,I)-AUX(N+4,I))*H
      AUX(N-4,I)=.390625E-2*(12.00*AUX(N-1,I)+135.00*AUX(N-2,I)+
     1108.00*AUX(N-3,I)+AUX(N-4,I))-.0234375  *(AUX(N+6,I)+
     218.00*AUX(N+5,I)-9.00*AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  225 Y(I)=AUX(N-4,I)
      DELT=DELT-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(MM.GT.1) RETURN
      DO 226 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
      AUX(16,I)=8.96296296*(AUX(N-1,I)-Y(I))-3.36111111*H*(AUX(N+6,I)+
     1DELT+DERY(I))
  226 AUX(N+3,I)=DERY(I)
      GO TO 206
      END
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESSURE DISTRI
C     BUTIONS FOR FREE STEAM MACH NUMBER M ABOVE THE UPPER CRITICAL ON
C     NONLIFTING BODIES OF REVOLUTION HAVING ORDINATES R PROPORTIONAL
C     TO X-X**N OR 1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZA-
C     TION--FOR REFFRENCE SEE SPREITER,J.R. AND ALKSNE,A.Y.,NASA TR-R2
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--LOCATION AS FRACTION OF BODY LENGTH (X/L) OF
C     POSITION OF MAXIMUM BODY THICKNESS
C     COLUMNS 21 TO 30--FINENESS RATIO OF BODY
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 50--INDEX FOR FLOW FIELD PRINT-OUT***INDEX EQUAL TO
C     OR LESS THAN 0.0 FOR NO FLOW FIELD CALCULATION AND GREATER THAN
C     0.0 FOR FLOW FIELD CALCULATION
C*********************************************************************
C
C
C     MAIN PROGRAM
      REAL M
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION XMTKS(19),DNEXP(19),DZST(19)
      EXTERNAL FCT,OUTP
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /C/ DCTAU2,DN
      COMMON /F/ DZS
      COMMON /L/ K
      COMMON /P/ DC
      COMMON /MRMC/ DWRTE,DMRM,FLWFLD
      DATA XMTKS/0.05,.10,.15,.20,.25,.30,.35,.40,.45,.50,.55,.60,.65,
     1          .70,.75,.80,.85,.90,.95/
      DATA DNEXP/88.731,34.649,19.173,12.215,8.396,6.044,4.482,3.389,
     1           2.595,2.0,2.595,3.389,4.482,6.044,8.396,12.215,19.173,
     2           34.649,88.731/
      DATA DZST/0.0077126,0.019182,0.033579,0.050724,0.070564,0.093097,
     1          0.118356,0.146406,0.177347,0.211325,0.272310,0.337355,
     2          0.406294,0.479059,0.555659,0.636159,0.720673,0.809353,
     3          0.902386/
   10 READ (5,200) M,XMT,F,DX,FLWFLD
  200 FORMAT (5F10.5)
      WRITE (6,600)
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LT.0.) GO TO 304
      IF(M.LT.1.) GO TO 306
      IF(DX.LE.0..OR.DX.GT.1.) GO TO 307
      DMM=DX
      DM=M
      GAMMA=1.4
      K=1
      IF (XMT.GT.0.5) K=0
      DO 913 I=1,19
      J=I-1
      IF(XMTKS(I)-XMT) 913,914,951
  913 CONTINUE
  914 DN=DNEXP(I)
      DZS1=DZST(I)
      GO TO 950
  951 IF (I.EQ.1) GO TO 914
      DN=DNEXP(J)+(DNEXP(I)-DNEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
      DZS1=DZST(J)+(DZST(I)-DZST(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
  950 CONTINUE
      A=XMT
      IF(XMT.LT..5) A=1.-XMT
      NIT=0
    5 DN1=DN-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**DN1
      DNUM=1.-DN*AN1
      DENOM=-AN1*(DN*ALOG(A)+1.)
      DNEWN=DN-(DNUM/DENOM)
      ERR=ABS(DN/DNEWN-1.)
      DN=DNEWN
      IF(ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (4X49HEXECUTION TERMINATED BECAUSE EXPONENT N CANNOT BE/5X
     149HDETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 10
    8 MM=1
      DC=(DN**(DN/(DN-1.00)))/(DN-1.00)
      DTAU=1./F
      DCTAU=DC*DTAU
      DCTAU2=DCTAU*DCTAU
      DZ0=1.E-2
      DZF=99.E-2
      D1M=DM*DM-1.
      DK=DM*DM*2.4
C
C      CALCULATION OF THE POINT WHERE S--(X) = 0
      N=1
    1 DZS=DZS1-DS2PI(DZS1)/DS3PI(DZS1)
      IF(ABS(DZS-DZS1).LT.1.E-6) GO TO 3
      DZS1=DZS
      N=N+1
      IF(N.GT.11) GO TO 2
      GO TO 1
    2 WRITE (6,100)
  100 FORMAT (1H0,4X55HEXECUTION TERMINATED BECAUSE S--(X) = 0 POINT CAN
     1NOT BE /5X57HDETERMINED TO WITHIN SUFFICIENT ACCURACY IN 10 ITERAT
     2IONS)
C
C     START OF INTEGRATION PROCEDURE
C
    3 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) DN
      WRITE (6,605) DZS
      WRITE (6,606) GAMMA
      WRITE (6,607) M
      DMRM=0.0
      IF (DMM-0.001) 920,920,921
  921 DWRTE=DZS/DMM + 1.0
      MRM=DWRTE
      DWRTE=MRM
      DWRTE=DWRTE*DMM + .001
      DMRM=-DMM
  920 CONTINUE
      WRITE(6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO NOSE//)
      IF (FLWFLD) 30,30,31
   30 WRITE (6,102)
  102 FORMAT (5X,1HX,7X8HCP(BODY)//)
      GO TO 32
   31 WRITE (6,112)
  112 FORMAT (5X,1HX,7X8HCP(BODY),5X6HCP(1D),6X6HCP(2D),6X,6HCP(3D),
     1        6X6HCP(4D),6X6HCP(5D),6X6HCP(6D),6X6HCP(7D),6X6HCP(8D)//)
   32 CONTINUE
      DY(1)=0.
      DDERY(1)=1.
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
   24 DPRMT(3)=-1.0E-2
      DPRMT(4)=1.E-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
```

```
      DY(1)=0.
      DDERY(1)=1.
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=1.0E-3
      DPRMT(4)=1.E-6
      NDIM=1
      IF (DMRM.EQ.0.0) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
      WRITE(6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO TAIL//)
      IF (FLWFLD) 40,40,41
   40 WRITE (6,102)
      GO TO 42
   41 WRITE (6,112)
   42 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      GO TO 10
  600 FORMAT (1H1,35X61HCALCULATION OF SURFACE AND FLOW FIELD PRESSURE
     1DISTRIBUTIONS / 36X65HFOR PURELY SUPERSONIC FLOW ABOUT A NONLIFTIN
     2G BODY OF REVOLUTION / 36X58HHAVING ORDINATES R PROPORTIONAL TO X-
     3X**N OR 1-X-(1-X)**N / 36X42HBY USING THE METHOD OF LOCAL LINEARIZ
     4ATION ///)
  601 FORMAT (1H ,49HBODY OF REVOLUTION AND FLOW FIELD CHARACTERISTICS//
     1)
  602 FORMAT (1H ,29HBODY MAX. THICKNESS AT X/L = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,13X,E12.5)
  604 FORMAT (1H ,32HEXPONENT N FOR BODY ORDINATES = ,E12.5)
  605 FORMAT (1H ,20HS--(X) = 0 AT X/L = ,12X,E12.5)
  606 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,6X,E12.5)
  607 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,4X,E12.5)
  303 WRITE (6,400)
  400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
      GO TO 10
  304 WRITE (6,401)
  401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
      GO TO 10
  306 WRITE (6,402)
  402 FORMAT(1/100HFREE STREAM MACH NUMBER MUST BE GREATER THAN THE UPPER
     1CRITICAL MACH NUMBER WHICH IS GREATER THAN 1  )
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (/89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT MUS
     1T BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END



      SUBROUTINE FCT(DX,DY,DZ)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /E/ DHF
      COMMON /F/ DZS
      DA=DSPI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      DHF=DA2*ALOG(DA/(DX*DX))+2.00*DINT(DX)
      DUU=D1M+DK*(DY(1)+DHF)
      IF (DUU) 2,2,1
    1 DZ(1)=DA3*ALOG(DUU)
      RETURN
    2 MM=2
      WRITE (6, 700) DX
  700 FORMAT (1H0,42HLOG ARGUMENT (MM-1+KU) IS NEGATIVE AT X = ,E12.5)
      IF(DX-DZS) 3,4,4
    3 WRITE (6,701)
  701 FORMAT (1H ,69HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER LESS T
     1HAN UPPER CRITICAL)
    4 RETURN
      END



      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
      DIMENSION DY(1),DDERY(1),DPRMT(5)
      DIMENSION DCPF(8)
      COMMON /E/ DHF
      COMMON /WRTC/ DWRTE,DMRM,FLWFLD
      DU=DY(1)+DHF
      DCP=-2.00*DU-DRSQ(DX)
      IF (IHLF.GT.10) GO TO 10
      IF (DX.LE.0.01 .OR. DX.GE.0.99) GO TO 20
      IF (DMRM) 50,20,60
   50 IF (DX.GT.DWRTE) RETURN
      GO TO 20
   60 IF (DX.LT.DWRTE) RETURN
   20 IF (FLWFLD) 30,30,31
   30 WRITE (6,101) DX,DCP
      GO TO 32
   31 D1=-2.0*DU
      D2=-DRSQ(DX)
      D3=-4.*DS2PI(DX)
      D4=DRRR(DX)
      DO 7 I=1,8
      DA=I
      D8=DA*D4
    7 DCPF(I)=D1+D2/(D3*DF)+D3*ALOG(D4)
      WRITE (6,101) DX,DCP,DCPF
  101 FORMAT (1X,F8.5,1X1PE11.4,1X,8(1X)PE11.4))
   32 CONTINUE
      DWRTE=DWRTE+DMRM
      RETURN
   10 WRITE (6,100)
  100 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      GO TO 2
      END



      FUNCTION DINT(DZ)
      EXTERNAL DFUN
      COMMON /G/ DW,DA2
      DW=DZ
      DA2=DS2PI(DZ)
      NIT=10
      DTOL=1.E-6
      CALL SIMP(0.00,DZ,NIT,NIT1,DTOL,DFUN,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT=DANS
      RETURN
   20 WRITE(6,21)
   21 FORMAT (1H0,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
     1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
     2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X,  40HFOR THE GIVEN NUMBE
     3R OF ITERATIONS (NIT))
      WRITE (6,22) DZ
   22 FORMAT (1H ,5X,4HX = ,E14.7)
      WRITE (6,23) DTOL
   23 FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,E14.7)
      STOP
      END
```

```
      FUNCTION DFUN(DZ)
      COMMON /G/ DW,DA2
      IF(ABS(DZ-DW).LT.1.E-6)GO TO 20
      DB2=DS2PI(DZ)
      DFUN=(DA2-DB2)/(DW-DZ)
      RETURN
   20 DFUN=DS3PI(DW)
      RETURN
      END
```

```
      FUNCTION DSPI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ-DZ**DN
    1 DSPI=DCTAU2*DS1*DS1/16.
      RETURN
    2 DS1=1.00-DZ-(1.00-DZ)**DN
      GO TO 1
      END
```

```
      FUNCTION DS2PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.)
    1 DS2PI=DCTAU2*(1.00-DN*DS1*(DN+1.00-(2.00*DN-1.00)*DS1))/8.
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      GO TO 1
      END
```

```
      FUNCTION DS3PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(-DN-1.00+2.00*(2.00*DN-1.00)*DS1
     1*DZ))/8.
      RETURN
    2 DS1=(1.00-DZ)**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(DN+1.00-2.D0*(2.00*DN-1.00)*DS1*
     1(1.00-DZ)))/8.
      RETURN
      END
```

```
      FUNCTION DRSQ(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
      DS2=1.00-DN*DS1
    1 DRSQ=DCTAU2*DS2*DS2/4.
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      DS2=-1.00+DN*DS1
      GO TO 1
      END
```

```
      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      NIT1=10
      DH=(DXU-DXL)/2.
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.00*DSUM2)/3.
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.
      DSUM3=0.
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DK=K
    2 DSUM3=DSUM3+FUN(DXL+DH*DK)
      DANS=DH*(DSUM1+2.00*DSUM2+4.00*DSUM3)/3.
      DERR=(DANS-DANS1)
      IF(ABS(DERR).LE.DTOL) RETURN
    1 DSUM2=DSUM2+DSUM3
      DTOL=DERR
      NIT1=0
      RETURN
      END
```

```
      FUNCTION DRRB(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      COMMON /P/ DC
      IF (K.GT.0) GO TO 2
      DA=DC*(DC-DZ**DN)
    1 DRRB=2./DA
      RETURN
    2 DA=DC*(1.-DZ-(1.-DZ)**DN)
      GO TO 1
      END
```

```
      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE HPCG (SINGLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     THE SURFACE AND FLOW FIELD PRESSURE DISTRIBUTION FOR FREE STREAM MACH
C     NUMBERS BELOW THE LOWER CRITICAL ON NONLIFTING BODIES OF REVOLUTION
C     HAVING ORDINATES PROPORTIONAL TO 1-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESSURE DISTRIBUTION
C     FOR FREE STREAM MACH NUMBERS AT OR NEAR 1 ON NONLIFTING BODIES OF
C     REVOLUTION HAVING ORDINATES PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     BY USING THE METHOD OF LOCAL LINEARIZATION--FOR [illegible] SEE
C     SPREITER,J.R AND ALKSNE,A.Y., NASA TR-R2
C
C
C*****************************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--LOCATION AS FRACTION OF BODY LENGTH (X/L) OF
C     POSITION OF MAXIMUM BODY THICKNESS
C     COLUMNS 21 TO 30--FINENESS RATIO OF BODY
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 50--INDEX FOR FLOW FIELD PRINT-OUT***[illegible] EQUAL TO
C     OR LESS THAN 0.0 FOR NO FLOW FIELD CALCULATION AND [illegible] THAN
C     0.0 FOR FLOW FIELD CALCULATION
C*****************************************************************************
C
C
C     MAIN PROGRAM
      DOUBLE PRECISION DPRMT,DY,DDERY,DAUX
      DOUBLE PRECISION DM,DK,D1,D1M,D2[illegible],DZ3,DU1,DU2,DU3,DU4,DU5
      DOUBLE PRECISION DX11,DX12,DUU1,DUU2,DCTAU2,D2N,[illegible],DTOL
      DOUBLE PRECISION DA,DA1,DA2,DA3,DA4,DA5,DA6,DA7,[illegible],D1N,D3N,D4N,
     1D1N2,D1N3,DSPI,DS1PI,DS2PI,DS3PI,DS4PI,DS5PI,DS6PI,DS7PI,DLOG
      DOUBLE PRECISION DU,DINT,D1I,D2I,D3I,DUI1,D4PC,[illegible],D4M1
      DOUBLE PRECISION D21,D2C,D241,D311,D3C1
      DOUBLE PRECISION D31,D41,D3C,D411,D4C1
      DOUBLE PRECISION D41,D51,D511,D4C
      DOUBLE PRECISION DX1,DHF,DSL,D[illegible],DX3,DX4,DINT1
      DOUBLE PRECISION DIB5,DIB4,DIB3,DIF2,DIB1,DIF0,DIF1,DIF2,DIF3,
     1DIF4,DIF5
      DOUBLE PRECISION DC,DN,DT,DF,DZS1,DTAU,DZO,DZF,DH,DCTAU
      DOUBLE PRECISION DH1,DF1
      DOUBLE PRECISION DNUM3,DNUM5,DNUM4
      DOUBLE PRECISION XMT,F,DMM,DX,GAMMA,A,DN1,DNUM,
     1DENOM,AN1,DNEWN,ERR,DWRTE,DMRM,FLWFLD
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION XMTKS(19),DNEXP(19),DZST(19)
      EXTERNAL FCT,OUTP
      COMMON /A/ DM,DK,D1,D1M,D2N,DZ[illegible],DU1,DU2,DU3,DU4,DU5
      COMMON /B/ M[illegible]
      COMMON /BB/ [illegible]
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /E/ DX11,DX12,DUU1,DUU2
      COMMON /F/ DV,DV3
      COMMON /H/ DTOL
      COMMON /L/ K
      COMMON /P/ DC
      COMMON /WRMC/ DWRTE,DMRM,FLWFLD
      DATA XMTKS/0.05,.10,.15,.20,.25,.30,.35,.40,.45,.50,.55,.60,.65,
     1           .70,.75,.80,.85,.90,.95/
      DATA DNEXP/88.731,34.54[illegible],17.17[illegible],7.21[illegible],[illegible],6.0[illegible],4.4[illegible],3.389,
     1           2.595,2.0,2.595,3.3[illegible],4.472,6.[illegible],[illegible],17.215,19.173,
     2           34.649,88.731/
      DATA DZST/0.0077126,0.0[illegible]182,0.[illegible]357[illegible],0.[illegible],0.[illegible],0.093[illegible]7,
     1           0.118356,0.146406,0.177[illegible]47,0.2[illegible]75,0.272310,0.337395,
     2           0.406234,0.47[illegible],[illegible],[illegible],[illegible]353,
     3           0.902386/
   10 READ (5,100) DM,XMT,F,D[illegible],FLWFLD
  100 FORMAT (5F10.5)
      WRITE (6,600)
  600 FORMAT (1H1,25X61HCALCULATION OF SURFACE AND FLOW FIELD PRESSURE
     1DISTRIBUTIONS /36X53HFOR FREE [illegible] AT [illegible] 1 ON
     2 A NONLIFTING /36X60HBODY OF [illegible]
     3TIONAL TO X-X**N /36X58H[illegible] 1-X-(1-X)**N [illegible] THE [illegible] OF LOC
     4AL LINEARIZATION ///)
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LT.0.) GO TO 304
      IF(DX.LT.0..OR.DX.GT.1.) GO TO 307
      DMM=DX
      K=1
      IF (XMT.GT.0.5) K=0
      DO 913 I=1,19
      J=I-1
      IF (XMTKS(I)-XMT) 913,914,951
  913 CONTINUE
  914 DN=DNEXP(I)
      DZS1=DZST(I)
      GO TO 950
  951 IF (I.EQ.1) GO TO 914
      DN=DNEXP(J)+(DNEXP(I)-DNEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
      DZS1=DZST(J)+(DZST(I)-DZST(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
  950 CONTINUE
      A=XMT
      IF(XMT.LT..5) A=1.-XMT
      NIT=
    5 DN1=DN-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**DN1
      DNUM=1.-DN*AN1
      DENOM=-AN1*(DN*DLOG(A)+1.)
      DNEWN=DN-(DNUM/DENOM)
      ERR=DABS(DN/DNEWN-1.)
      DN=DNEWN
      IF(ERR.GT.1.D-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (4X49HEXECUTION TERMINATED BECAUSE EXPONENT N CANNOT BE /5X
     149HDETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 10
    8   =1
      DC=(DN**(DN/(DN-1.D0)))/(DN-1.D0)
      DTAU=1./F
      DCTAU=DC*DTAU
      DCTAU2=DCTAU*DCTAU
      GAMMA=1.4
      D2N=2.D0*DN
      DK=DM*DM*2.4D0
      D1=DK*1.781072417990198D0
      D1M=(DM*DM-1.D0)
      D2M=D1M/DK
      DTOL=1.D-12
      DH=1.D-3
      DZO=.05D0
      DZF=.95D0
      DX12=0.D0
      D[illegible]2=0.D0
      M=1
      [illegible]=1
C
C     CALCULATION OF THE POINT WHERE S--(X) = 0
C
      N=1
    1 DZS=DZS1-DS2PI(DZS1)/DS3PI(DZS1)
      IF (DABS(DZS-DZS1).LT.1.D-12) GO TO 3
      DZS1=DZS
      N=N+1
      IF(N.GT.10) GO TO 2
      GO TO 1
    2 WRITE (6,101)
  101 FORMAT (1H0,4X55HEXECUTION TERMINATED BECAUSE S--(X) = 0 POINT CAN
     1NOT BE /5X57HDETERMINED TO WITHIN SUFFICIENT ACCURACY IN 10 ITERAT
     2IONS)
      GO TO 10
    3 DZS1=DZS
       WRITE (6,601)
       WRITE (6,602) XMT
       WRITE (6,6[illegible])
       WRITE (6,6[illegible])
```

```
      WRITE (6,605) DZS
      WRITE (6,606) 
      WRITE (6,607) 
      DNUM=0.0
      IF (DNUM-0.001) 927,928,
  921 DWRTE=DZS/  / + 1.0
      
      DWRTE=
      DWRTE=DWRTE*D  + .001
      DNUM=-D 
  920 CONTINUE
C
C     CALCULATION OF AREA AND DERIVATIVES OF AREA AT XS
C
    4 DA=DSPI(DZS)
      DA1=DS1PI(DZS)
      DA2=DS2PI(DZS)
      DA3=DS3PI(DZS)
      DA4=DS4PI(DZS)
      DA5=DS5PI(DZS)
      DA6=DS6PI(DZS)
      DA7=DS7PI(DZS)
      DAM=DLOG(D1*DA/DZS)
      D1N=DA1/DA
      D3N=DA3/DA
      D4N=DA4/DA
      D1N2=D1N*D1N
      D1N3=D1N*D1N2
      DIB5=DINT(DZS-5.D0*DH)
      DIB4=DINT(DZS-4.D0*DH)
      DIB3=DINT(DZS-3.D0*DH)
      DIB2=DINT(DZS-2.D0*DH)
      DIB1=DINT(DZS-DH)
      DIB0=DINT(DZS)
      DIF1=DINT(DZS+DH)
      DIF2=DINT(DZS+2.D0*DH)
      DIF3=DINT(DZS+3.D0*DH)
      DIF4=DINT(DZS+4.D0*DH)
      DIF5=DINT(DZS+5.D0*DH)
C
C     CALCULATION OF INITIAL VALUE OF U
C
      DU=-D1M+DINT(DZS)
C
C     CALCULATION OF THE 1ST DERIVATIVE OF U
C
      D1I=(45.D0*(DIF1-DIB1)-9.D0*(DIF2-DIB2)+DIF3-DIB3)/(60.D0*DH)
      DU1=5.D-1
      N=1
   13 DU11=DU1-DU1*(DU1-DA3*(DLOG(DU1)+DAM)-D1I)/(DU1-DA3)
      IF (DABS(DU11-DU1).LT.1.D-10) GO TO 14
      DU1=DU11
      N=N+1
      IF(N.GT.11) GO TO 200
      GO TO 13
   14 DU1=DU11
  200 WRITE (6,104)
  104 FORMAT (1H0, 4X73HPROGRAM TERMINATED BECAUSE INITIAL DERIVATIVE OF
     1 U AT S--(X) = 0 REQUIRED /5X68HFOR TAYLOR SERIES START AT THAT PO
     2INT CANNOT BE DETERMINED TO WITHIN /5X36HSUFFICIENT ACCURACY IN 10
     3 ITERATIONS)
      IF(N.GT.11) GO TO 10
C
C     CALCULATION OF THE INITIAL 2ND  ERIVATIVE OF U
C
      D2I=(-490.D0*DIB0+270.D0*(DIF1+DIB1)-27.D0*(DIF2+DIB2)+2.D0*(DIF3+
     1DIB3))/(180.D0*DH*DH)
      DA31=DA3/DU1
      DAM1=DAM+DLOG(DU1)
      DU2=(DA4*DAM1+2.D0*DA3*(D1N*(1.D0-DA31)-1.D0/DZS)+D2I)/(1.D0-2.D0
     1*DA31)
C
C     CALCULATION OF THE INITIAL 3RD  ERIVATIVE OF U
C
      DNUM3=(-488.D0*(DIF1-DIB1)+338.D0*(DIF2-DIB2)-72.D0*(DIF3-DIB3)+
     17.D0*(DIF4-DIB4))
      IF(DABS(DNUM3).LT.1.D-12) GO TO 117
      D3I=DNUM3/(240.D0*DH*DH*DH)
      GO TO 116
  117 D3I=0.D0
  116 D21=DU2/DU1-DA31*D1
      D2C=D21+D1 -1.D0/DZS
      DA41=DA4/DU1
      D311=D1 *(D 41-2.D0*DA31*D1N)
      D3C1=-D21*D21-D1N2+1.D0/(DZS*DZS)
      DU3=(DA5*DAM1+3.D0*DA4*D2C+3.D0*DA3*(D3C1-D311)+D3I)/(1.D0-3.D0*
     1DA31)
C
C     CALCULATION OF THE INITIAL 4TH  ERIVATIVE OF U
C
      DNUM4=(2730.D0*DIB0-1   .D0*(DIF1+DIB1)+676.D0*(DIF2+DIB2)-96.D0*
     1(DIF3+DIB3)+7.D0*(DIF4+DIB4))
      IF(DABS(DNUM4).LT.1.D-12)    TO 118
      D4I=DNUM4/(240.D0*DH*DH*DH*  )
      GO TO 119
  118 D4I=0.D0
  119 D31=DU3/DU1-D311
      D3C=D31+D3C1
      D411=(D1N*(DA5-3.D0*DA4*D1 )+D3N*(6.D0*DA1*D1N2+DA3))/DU1
      D4C1=-3.D0*D31*D21*  2)+2.D0*(D21**3)+D3N+2.D0* 1N2-2.D0/(DZS**3)
      DU4=(DA6*DAM1+4.D0*DA5*D2C+ .D0*DA4*D3C+4.D0*DA3*(-D411+D4C1)+D4I
     1)/(1.D0-4.D0*DA31)
C
C     CALCULATION OF THE INITIAL 5TH  ERIVATIVE OF U
C
      DNUM5=(1938.D0*(DIF1-DIB1)-1872.D0*(DIF2-DIB2)+753.D0*(DIF3-DIB3)
     1-152.D0*(DIF4-DIB4)+13.D0*(DIF5-DIB5))
      IF(DABS(DNUM5).LT.1.D-12) GO TO 120
      D5I=DNUM5/(288.D0*(DH**5))
      GO TO 121
  120 D5I=0.D0
  121 D511=(D1N*(DA6-4.D0*DA5*D1N)+4.D0*DA4*D1N3+(D3N+2.D0*D1 3)*(4.D0
     1*DA4-12.D0*DA3*D1N))/DU1
      D41=DU4/DU1-D411
      D4C=D41+D4C1
      DU5=(DA7*DAM1+5.D0*DA6*D2C+10.D0*DA5*D3C+10.D0*DA4*D4C+5.D0*DA3*
     1(-D511-4.D0*D41*D21+12.D0*D31*D21*D21-3.D0*D31*D31-6.D0*(D21**4)+
     2D4N-4.D0*D3N*D1N-6.D0*D1N2+6.D0/(DZS**4))+D5I)/(1.D0-5.D0*DA31)
      DTOL=1.D-8
C
C
C     START OF INTEGRATION PROCEDURE
      WRITE (6,99)
   99 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO NOSE  //)
      IF(FLMFLD) 60,60,61
   60 WRITE (6,102)
  102 FORMAT (5X,1HX,10X,8HCP(BODY)//)
      GO TO 62
   61 WRITE (6,103)
  103 FORMAT (1H0,5X,1HX,9X,8HCP(BODY),7X,6HCP(1D),6X,6HCP(2D),6X,6HCP(3
     1D),6X,6HCP(4D),6X,6HCP(5D),6X,6HCP(6D),6X,6HCP(7D),6X,6HCP(8D)//)
   62 CONTINUE
      DY(1)=DU
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZ.
   24 DPRMT(3)=-1.D-3
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,  ,  ED ,  I ,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
  700 WRITE (6,98)
   98 FORMAT (1H0,44HSTART OF INTEGRATION FROM S--(X) = 0 TO TAIL//)
```

```
      IF(FLWFLD) 70,70,71
   70 WRITE (6,102)
      GO TO 72
   71 WRITE (6,103)
   72 CONTINUE
   29 DPRMT(1)=DZS
      DPRMT(2)=DZF
      DY(1)=DU
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
   27 DPRMT(3)=1.D-3
      DPRMT(4)=1.D-6
      DDERY(1)=1.D0
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.EQ.2) GO TO 29
      IF(MM.EQ.3) GO TO 30
   28 GO TO 10
   29 DY(1)=DV
      DPRMT(1)=DV3
      MM=1
      IF(FLWFLD) 40,40,41
   40 WRITE (6,102)
      GO TO 27
   41 WRITF (6,103)
      GO TO 27
   30 DSL=(DX12-DX11)/(DUU1-DUU2)
      DSH=DSL*(DUU2+D2M)
      DX3=DX12+DSH
      DX4=2.D0*DX3-DX12
      WRITE(6,205)
  205 FORMAT(1H0,29HSTART OF SUBSONIC CALCULATION)
      WRITE (6,509) DX3
  509 FORMAT (1H0,35HLOG ARGUMENT (M*M-1+KU) = 0 AT X = ,D12.5)
      WRITE (6,510) DX4
  510 FORMAT (1H0,35HSUBSONIC CALCULATION BEGINS AT X = ,D12.5)
      DY(1)=-DUU2-DS2PI(DX4)*DLOG(DSPI(DX4)/(DX4*(1.D0-DX4)))-DINT(DX4)
     1-DINT1(DX4)-2.D0*D2M
      DPRMT(1)=DX4
      MM=1
      IF(FLWFLD) 42,42,43
   42 WRITE (6,102)
      GO TO 27
   43 WRITE (6,103)
      GO TO 27
  303 WRITE (6,400)
  400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
      GO TO 10
  304 WRITF (6,401)
  401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (/89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT MUS
     1T BE GREATER THAN 0 AND LESS THAN 1  )
      GO TO 10
  601 FORMAT (1H ,49HBODY OF REVOLUTION AND FLOW FIELD CHARACTERISTICS//
     1)
  602 FORMAT (1H ,29HBODY MAX. THICKNESS AT X/L = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,13X,E12.5)
  604 FORMAT (1H ,32HEXPONENT N FOR BODY ORDINATES = ,E12.5)
  605 FORMAT (1H ,20HS--(X) = 0 AT X/L = ,12X,E12.5)
  606 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,6X,F12.5)
  607 FORMAT (1H ,28HFREE STREAM MACH NUMBER M = ,4X,E12.5)
      END
```

```
      SUBROUTINE FCT(DX,DY,DZ)
      DOUBLE PRECISION DX,DY,DZ,DV,DV3,DHF,DEF,DABS
      DOUBLE PRECISION DM,DK,D1,D1M,D2M,DZS,DU1,DU2,DU3,DU4,DU5
      DOUBLE PRECISION DA1,DA2,DA3,DA,DSPI,DS1PI,DS2PI,DS3PI
      DOUBLE PRECISION DEXP,DINT,DLOG,DUU,DV1,DV2,DZH,D1INT,DIFF,DINT1
      COMMON /A/ DM,DK,D1,D1M,D2M,DZS,DU1,DU2,DU3,DU4,DU5
      DIMENSION DY(1),DZ(1)
      COMMON /B/ MM
      COMMON /E/ M
      COMMON /F/ DV,DV3
      COMMON /G/ DHF,DEF
      IF(DABS(DX-DZS).LE.5.D-3) GO TO 10
      DA=DSPI(DX)
      DA1=DS1PI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      IF (M.EQ.1) GO TO 1
      IF(M.EQ.2) GO TO 2
      IF(M.EQ.3) GO TO 3
    1 IF(DX-DZS) 20,20,30
   20 DZ(1)=DA1*DA2/DA+DEXP((DY(1)+D2M-DINT(DX)-DA2*DLOG(D1*DA/DX))/DA2)
      RETURN
   30 DUU=D1M+DK*DY(1)
      IF(DUU) 20,20,40
   40 DV1=DA/(DX*DX)
      DV2=DINT(DX)
      DZH=DA3*DLOG(DUU*DV1)+DA2*(DA1/DA-2.D0/DX)+2.D0*D1INT(DX)
      DZ(1)=DA1*DA2/DA+DEXP((DY(1)+D2M-DV2-DA2*DLOG(D1*DA/DX))/DA2)
      DIFF=DZH-DZ(1)
      IF(DIFF) 60,60,50
   50 RETURN
   60 M=2
      MM=2
      WRITE (6,199)
  199 FORMAT (1H0,31HSTART OF SUPERSONIC CALCULATION)
      WRITE (6,200) DX
  200 FORMAT (1H0, 37HSUPERSONIC CALCULATION STARTS AT X = ,D12.5//)
      DV=DY(1)-DA2*DLOG(DV1)-2.D0*DV2
      DV3=DX
      RETURN
    2 DHF=DA2*DLOG(DA/(DX*DX))+2.D0*DINT(DX)
      DUU=D1M+DK*(DY(1)+DHF)
      IF(DUU) 80,80,70
   70 DZ(1)=DA3*DLOG(DUU)
      RETURN
   80 M=3
      MM=3
      RETURN
    3 DEF=DA2*DLOG(DA/((DX*(1.D0-DX))))+DINT(DX)+DINT1(DX)
      DUU=-D1M-DK*(DY(1)+DEF)
      DZ(1)=DA3*DLOG(DUU)
      RETURN
   10 DZ(1)=DU1+DU2*(DX-DZS)+DU3*((DX-DZS)**2)/2.D0+DU4*((DX-DZS)**3)/
     16.D0+DU5*((DX-DZS)**4)/24.D0
      RETURN
      END
```

```
      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
      DOUBLE PRECISION DP
      DOUBLE PRECISION DX,DY,DDERY,DPRMT,DX11,DX12,DUU1,DUU2
      DOUBLE PRECISION DHF,DEF,DCP,DRRB,DU
      DOUBLE PRECISION DZS1,DB1,DF1,D1,D2,D3,D4,DCPB,DCPF,DRRB,DS2PI,
     1DLOG
      DOUBLE PRECISION DWRTE,DMRM,FL,FLD
      DIMENSION DY(1),DDERY(1),DPRMT(5),DCPF(8)
      COMMON /B/ MM
      COMMON /E/ M
      COMMON /F/ DX11,DX12,DUU1,DUU2
      COMMON /G/ DHF,DEF
```

```
      COMMON /WRITE/ DWRITE,DWRM,FLAFLG
      IF(IHLF.GT.10) GO TO 1
      IF(M.EQ.2) GO TO 3
   21 IF(DWRM) 60,20,60
   60 IF(DX.GT.DWRITE) RETURN
      GO TO 20
   60 IF(DX.LT.DWRITE) RETURN
   20 IF(M.EQ.2) GO TO 10
      IF(M.EQ.3) GO TO 11
      DU=DY(1)
   10 IF(FLAFLG) 30,30,31
   30 DCP=-2.D0*DU-DRSQ(DX)
      WRITE (6,101) DX,DCP
      GO TO 32
   31 D1=-2.D0*DU
      D2=-DRSQ(DX)
      D3=-4.D0*DS2PI(DX)
      D4=DRRR(DX)
      DCP=D1+D2
      DO 7 I=1,8
      DA=I
      DB=DA*D4
    7 DCPF(I)=D1+D2/(DB*DB)+D3*DLOG(DB)
      WRITE(6,101)DX,DCP,DCPF
  101 FORMAT (1X,D12.5,1X,D11.4,1X,8(1X,D11.4))
   32 CONTINUE
      DWRITE=DWRITE+DWRM
      RETURN
    3 DU=DY(1)+DHF
      DX11=DX12
      DX12=DX
      DUU1=DUU2
      DUU2=DU
      GO TO 21
   11 DU=DY(1)+DEF
      GO TO 10
    1 WRITE (6,100)
  100 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      IF(M.EQ.2) DU=DY(1)+DHF
      GO TO 20
      END
```

```
      FUNCTION DINT(DZ)
      DOUBLE PRECISION DINT,DZ,DFUN,DW,DA2,DTOL,DABS,DS2PI,DANS
      EXTERNAL DFUN
      COMMON /G/ DW,DA2
      COMMON /H/ DTOL
      IF(DABS(DZ).LT.1.D-6) GO TO 25
      DW=DZ
      DA2=DS2PI(DZ)
      NIT=10
      CALL SIMP(0.D0,DZ,NIT,NIT1,DTOL,DFUN,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT=DANS
      RETURN
   20 WRITE(6,21)
   21 FORMAT (1H0,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
     1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
     2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X, 40HFOR THE GIVEN NUMBE
     3R OF ITERATIONS (NIT))
      WRITE(6,22) DZ
   22 FORMAT(1H ,5X,4HX = ,D24.16)
      WRITE (6,23) DTOL
   23 FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,D24.16)
      STOP
   25 DINT=0.D0
      RETURN
      END
```

```
      FUNCTION DINT1(DZ)
      DOUBLE PRECISION DINT1,DZ,DFUN,DW,DA2,DTOL,DABS,DS2PI,DANS
      EXTERNAL DFUN
      COMMON /G/ DW,DA2
      COMMON /H/ DTOL
      IF(DABS(1.D0-DZ).LT.1.D-6) GO TO 25
      DW=DZ
      DA2=DS2PI(DZ)
      NIT=10
      CALL SIMP(1.D0,DZ,NIT,NIT1,DTOL,DFUN,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT1=DANS
      RETURN
   20 WRITE (6,21)
   21 FORMAT (1H0,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
     1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
     2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X, 40HFOR THE GIVEN NUMBE
     3R OF ITERATIONS (NIT))
      WRITE (6,22) DZ
   22 FORMAT (1H ,5X,4HX = ,D24.16)
      WRITE (6,23) DTOL
   23 FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,D24.16)
      STOP
   25 DINT1=0.D0
      RETURN
      END
```

```
      FUNCTION D1INT(DZ)
      DOUBLE PRECISION D1INT,DZ,DH,DINT,DIB3,DIB2,DIB1,DIF1,DIF2,DIF3
      DH=1.D-3
      DIB3=DINT(DZ-3.D0*DH)
      DIB2=DINT(DZ-2.D0*DH)
      DIB1=DINT(DZ-DH)
      DIF1=DINT(DZ+DH)
      DIF2=DINT(DZ+2.D0*DH)
      DIF3=DINT(DZ+3.D0*DH)
      D1INT=(45.D0*(DIF1-DIB1)-9.D0*(DIF2-DIB2)+DIF3-DIB3)/(60.D0*DH)
      RETURN
      END
```

```
      FUNCTION DFUN(DZ)
      DOUBLE PRECISION DFUN,DZ,DW,DA2,DABS,DB2,DS2PI,DS3PI
      COMMON /G/ DW,DA2
      IF(DABS(DZ-DW).LT.1.D-6) GO TO 20
      DB2=DS2PI(DZ)
      DFUN=(DA2-DB2)/(DW-DZ)
      RETURN
   20 DFUN=DS3PI(DW)
      RETURN
      END
```

```
      FUNCTION DRSQ(DZ)
      DOUBLE PRECISION DCTAU2,DN,D2N,DRSQ,DS1,DS2,DZ
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-1.D0)
      DS2=1.D0-DN*DS1
    1 DRSQ=DCTAU2*DS2*DS2/4.D0
      RETURN
    2 DS1=(1.D0-DZ)**(DN-1.D0)
      DS2=-1.D0+DN*DS1
      GO TO 1
      END
```

```
      FUNCTION DSPI(DZ)
      DOUBLE PRECISION DSPI,DZ,DCTAU2,DN,DS1,D2N
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ-DZ**DN
    1 DSPI=DCTAU2*DS1*DS1/16.D0
      RETURN
    2 DS1=1.D0-DZ-((1.D0-DZ)**DN)
      GO TO 1
      END
```

```
      FUNCTION DS1PI(DZ)
      DOUBLE PRECISION DCTAU2,DN,D2N,DS1,DS1PI,DZ
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-1.D0)
    1 DS1PI=DCTAU2*DZ*(1.D0-(DN+1.D0)*DS1+DN*DS1*DS1)/8.D0
      RETURN
    2 DS1=(1.D0-DZ)**(DN-1.D0)
      DS1PI=-DCTAU2*(1.D0-DZ)*(1.D0-(DN+1.D0)*DS1+DN*DS1*DS1)/8.D0
      RETURN
      END
```

```
      FUNCTION DS2PI(DZ)
      DOUBLE PRECISION DS2PI,DZ,DCTAU2,DN,DS1,D2N
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.D0)
    1 DS2PI=DCTAU2*(1.D0-DN*DS1*(DN+1.D0-(D2N-1.D0)*DS1))/8.D0
      RETURN
    2 DS1=(1.D0-DZ)**(DN-1.D0)
      GO TO 1
      END
```

```
      FUNCTION DS3PI(DZ)
      DOUBLE PRECISION DS3PI,DZ,DCTAU2,DN,DS1,DS2,D2N
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-2.D0)
      DS2=DZ**(D2N-3.D0)
    1 DS3PI=DCTAU2*DN*(DN-1.D0)*(-(DN+1.D0)*DS1+2.D0*(D2N-1.D0)*DS2)
     1/8.D0
      RETURN
    2 DS1=-((1.D0-DZ)**(DN-2.D0))
      DS2=-((1.D0-DZ)**(D2N-3.D0))
      GO TO 1
      END
```

```
      FUNCTION DS4PI(DZ)
      DOUBLE PRECISION DS4PI,DZ,DCTAU2,DN,D2N,DS1,DS2
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-3.D0)
      DS2=DZ**(D2N-4.D0)
    1 DS4PI=DCTAU2*DN*(DN-1.D0)*(-(DN+1.D0)*(DN-2.D0)*DS1+2.D0*(D2N-
     1)*D0)*(D2N-3.D0)*DS2)/8.D0
      RETURN
    2 DS1=(1.D0-DZ)**(DN-3.D0)
      DS2=(1.D0-DZ)**(D2N-4.D0)
      GO TO 1
      END
```

```
      FUNCTION DS5PI(DZ)
      DOUBLE PRECISION DS5PI,DZ,DCTAU2,DN,D2N,DS1,DS2
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-4.D0)
      DS2=DZ**(D2N-5.D0)
    1 DS5PI=DCTAU2*DN*(DN-1.D0)*(DN-2.D0)*(-(DN+1.D0)*(DN-3.D0)*DS1+
     14.D0*(D2N-1.D0)*(D2N-3.D0)*DS2)/8.D0
      RETURN
    2 DS1=-((1.D0-DZ)**(DN-4.D0))
      DS2=-((1.D0-DZ)**(D2N-5.D0))
      GO TO 1
      END
```

```
      FUNCTION DS6PI(DZ)
      DOUBLE PRECISION DS6PI,DZ,DCTAU2,DN,D2N,DS1,DS2
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-5.D0)
      DS2=DZ**(D2N-6.D0)
    1 DS6PI=DCTAU2*DN*(DN-1.D0)*(DN-2.D0)*(-(DN+1.D0)*(DN-3.D0)*(DN-
     14.D0)*DS1+4.D0*(D2N-1.D0)*(D2N-3.D0)*(D2N-5.D0)*DS2)/8.D0
      RETURN
    2 DS1=(1.D0-DZ)**(DN-5.D0)
      DS2=(1.D0-DZ)**(D2N-6.D0)
      GO TO 1
      END
```

```
      FUNCTION DS7PI(DZ)
      DOUBLE PRECISION DS7PI,DZ,DCTAU2,DN,D2N,DS1,DS2
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      IF (K.GT.0) GO TO 2
      DS1=DZ**(DN-6.D0)
      DS2=DZ**(D2N-7.D0)
    1 DS7PI=DCTAU2*DN*(DN-1.D0)*(DN-2.D0)*(DN-3.D0)*(-(DN+1.D0)*(DN-
     14.D0)*(DN-5.D0)*DS1+8.D0*(D2N-1.D0)*(D2N-3.D0)*(D2N-5.D0)*DS2)/8.
     2D0
      RETURN
    2 DS1=-((1.D0-DZ)**(DN-6.D0))
      DS2=-((1.D0-DZ)**(D2N-7.D0))
      GO TO 1
      END
```

```
      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      DOUBLE PRECISION DH,DXU,DXL,DSUM1,DSUM2,DSUM3,DANS,DANS1,DF,
     1DABS,DERR,DTOL,FUN
      NIT1=10
      DH=(DXU-DXL)/2.D0
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.D0*DSUM2)/3.D0
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.D0
      DSUM3=0.D0
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DF=K
    2 DSUM3=DSUM3+FUN(DXL+DH*DF)
      DANS=DH*(DSUM1+2.D0*DSUM2+4.D0*DSUM3)/3.D0
      IF (DABS(DANS).LT.1.D-6) GO TO 1
      DERR=(DANS-DANS1)
      IF(DABS(DERR).LE.DTOL) RETURN
    1 DSUM2=DSUM2+DSUM3
      IF (DABS(DANS).LT.1.D-6) RETURN
      DTOL=DERR
      NIT1=0
      RETURN
      END



      FUNCTION DRRB(DZ)
      DOUBLE PRECISION DRRB,DN,D2N,DA,DC,DCTAU2,DZ
      COMMON /C/ DCTAU2,DN,D2N
      COMMON /L/ K
      COMMON /P/ DC
      IF(K.GT.0) GO TO 2
      DA=DC*(DZ-DZ**DN)
    1 DRRB=2.D0/DA
      RETURN
    2 DA=DC*(1.D0-DZ-(1.D0-DZ)**DN)
      GO TO 1
      END



      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
C
      DIMENSION PRMT(1),Y(1),DERY(1),AUX(16,1)
      DOUBLE PRECISION Y,DERY,AUX,PRMT,X,H,Z,DELT
      DOUBLE PRECISION DABS
      COMMON /B/ MM
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.D0
      DO 1 I=1,NDIM
      AUX(16,I)=0.D0
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C     ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C     COMPUTATION OF DERY FOR STARTING VALUES
    4 CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
C
C     RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C     COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.5D0*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(9,I)=DERY(I)
      ISW=3
      GO TO 100
C
C     COMPUTATION OF TEST VALUE DELT
   15 DELT=0.D0
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*DABS(Y(I)-AUX(4,I))
      DELT=.0666666666666667D0*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C     NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.
   18 IHLF=11
      X=X+H
      GO TO 4
C
C     THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
```

```
      220Y(I)=AUX(1,I)+H*(.375D0*AUX(8,I)+.791666666666667D0*AUX(9,I)
     1-.208333333333333D0*AUX(10,I)+.041666666666667D0*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+.333333333333333D0*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+.375D0*H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     ******************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+.4D0*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
      Z=X+.4D0*H
      CALL FCT(Z,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+.296977602477536D0*AUX(5,I)+.158759644971035BD0*Z
C
      Z=X+.455737254218789 4D0*H
      CALL FCT(Z,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+.218100388225920 5D0*AUX(5,I)-3.050965148692931D0*
     1AUX(6,I)+3.832864760467010D0*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 104 I=1,NDIM
 1040Y(I)=AUX(N,I)+.174602822626904D0*AUX(5,I)-.551480662878732 9D0*
     1AUX(6,I)+1.205535599396524D0*AUX(7,I)+.171184781219519 0D0*
     2H*DERY(I)
      GO TO(9,13,15,21),ISW
C     ******************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
```

```
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
     0DELT=AUX(N-4,I)+1.333333333333333D0*H*(AUX(N+6,I)+AUX(N+6,I)-
     1AUX(N+5,I)+AUX(N+4,I))
      Y(I)=DELT-.925619334710743 8D0*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
     0DELT=.125D0*(9.D0*AUX(N-1,I)-AUX(N-3,I)+3.D0*H*(DERY(I)+AUX(N+6,I)
     1+AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+.974380165289256 2D0*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0.D0
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*DABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF (MM.GT.1) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF(DABS(X-PRMT(2))-.1D0*DABS(H))212,215,215
  215 IF(DELT-.0200*PRMT(4))216,216,201
C
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
```

```
 2210AUX(16,I)=8.962962962962963D0*(Y(I)-AUX(N-3,I))
    1-3.361111111111111D0*H*(DERY(I)+DELT+AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=.5D0*H
      ISTEP=0
      DO 224 I=1,NDIM
     0Y(I)=.390625D-2*(8.D1*AUX(N-1,I)+135.D0*AUX(N-2,I)+4.D1*AUX(N-3,I)
     1+AUX(N-4,I))-.1171875D0*(AUX(N+6,I)-6.D0*AUX(N+5,I)-AUX(N+4,I))*H
     0AUX(N-4,I)=.390625D-2*(12.D0*AUX(N-1,I)+135.D0*AUX(N-2,I)+
     1108.D0*AUX(N-3,I)+AUX(N-4,I))-.0234375D0*(AUX(N+6,I)+
     218.D0*AUX(N+5,I)-9.D0*AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  225 Y(I)=AUX(N-4,I)
      DELT=DELT-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF (MM.GT.1) RETURN
      DO 226 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
     0AUX(16,I)=8.962962962962963D0*(AUX(N-1,I)-Y(I))
     1-3.361111111111111D0*H*(AUX(N+6,I)+DELT+DERY(I))
  226 AUX(N+3,I)=DERY(I)
      GO TO 206
      END
```

```
C     PROGRAM FOR DETERMINING THE LOWER CRITICAL MACH NUMBER ON NONLIFTING
C     BODIES OF REVOLUTION HAVING ORDINATES R PROPORTIONAL TO X-X**N OR
C     1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION--FOR REFERENCE
C     SEE SPREITER,J.R. AND ALKSNE,A.Y.,NASA TR-R2
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--INITIAL ESTIMATE OF LOWER CRITICAL MACH NUMBER
C     COLUMNS 11 TO 20--LOCATION AS FRACTION OF BODY LENGTH (X/L) OF
C     POSITION OF MAXIMUM BODY THICKNESS
C     COLUMNS 21 TO 30--FINENESS RATIO OF BODY
C*********************************************************************
C
C
C     MAIN PROGRAM
      REAL M
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION XMTKS(19),DNEXP(19),DZST(19)
      EXTERNAL FCT,OUTP
      COMMON /A/ DIM,DK
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      COMMON /R/ DLASTX,DLASTU,DLMDAE
      COMMON /S/ DBIGX,DBIGU,DUDIFF
      DATA XMTKS/0.05,.10,.15,.20,.25,.30,.35,.40,.45,.50,.55,.60,.65,
     1           .70,.75,.80,.85,.90,.95/
      DATA DNEXP/88.731,34.649,19.173,12.215,8.396,6.044,4.482,3.389,
     1           2.595,2.0,2.595,3.389,4.482,6.044,8.396,12.215,19.173,
     2           34.649,88.731/
      DATA DZST/0.0077126,0.019182,0.033578,0.050724,0.070564,0.093097,
     1          0.118356,0.146406,0.177347,0.211325,0.272310,0.337355,
     2          0.406294,0.479059,0.555659,0.636159,0.720673,0.809353,
     3          0.902386/
C
C     READING AND PRINTING OF INPUT DATA
C
   10 READ (5,200) M,XMT,F
  200 FORMAT (3F10.2)
      WRITE (6,600)
  600 FORMAT (1H1,35X78HCALCULATION OF LOWER CRITICAL MACH NUMBER FOR A
     1NONLIFTING BODY OF REVOLUTION /36X71HHAVING ORDINATES R PROPORTION
     2AL TO X-X**N OR 1-X-(1-X)**N BY USING THE /36X29HMETHOD OF LOCAL L
     3INEARIZATION ///)
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LE.0.) GO TO 304
      IF(M.GT.1..OR.M.LE.0.) GO TO 306
      DM=M
      K=1
      IF(XMT.GT..5) K=0
      DO 913 I=1,19
      J=I-1
      IF(XMTKS(I)-XMT) 913,914,951
  913 CONTINUE
  914 DN=DNEXP(I)
      DZS1=DZST(I)
      GO TO 950
  951 IF (I.EQ.1) GO TO 914
      DN=DNEXP(J)+(DNEXP(I)-DNEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
      DZS1=DZST(J)+(DZST(I)-DZST(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
  950 CONTINUE
      A=XMT
      IF(XMT.LT..5) A=1.-XMT
      NIT=0
    5 DN1=DN-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**DN1
      DNUM=1.-DN*AN1
      DENOM=-AN1*(DN*ALOG(A)+1.)
      DNEWN=DN-(DNUM/DENOM)
      ERR=ABS(DN/DNEWN-1.)
      DN=DNEWN
      IF(ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (4X49HEXECUTION TERMINATED BECAUSE EXPONENT N CANNOT BE /5X
     149HDETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 10
C
C     INITIALIZATION OF VARIOUS PARAMETERS
    8 DC=(DN**(DN/(DN-1.00)))/(DN-1.00)
      DMLOBD=-1.0
      DMUPBD=-1.0
      DTAU=1./F
      GAMMA=1.4
      NN=0
      DDM=.1
      DCTAU=DC*DTAU
      DCTAU2=DCTAU*DCTAU
      DZF=0.99
C
C      CALCULATION OF THE POINT WHERE S--(X) = 0
      N=1
    1 DZS=DZS1-DS2PI(DZS1)/DS3PI(DZS1)
      IF(ABS(DZS-DZS1).LT.1.E-6) GO TO 40
      DZS1=DZS
      N=N+1
      IF(N.GT.10) GO TO 3
      GO TO 1
    3 WRITE (6,100)
  100 FORMAT (1H0,4X55HEXECUTION TERMINATED BECAUSE S--(X) = 0 POINT CAN
     1NOT BE /5X57HDETERMINED TO WITHIN SUFFICIENT ACCURACY IN 10 ITERAT
     2IONS)
      GO TO 10
   40 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) DN
      WRITE (6,605) DZS
      WRITE (6,606) GAMMA
      WRITE (6,99)
   99 FORMAT (1H0,//)
      WRITE (6,110)
  110 FORMAT (1H0,9HITERATION,2X,11HMACH NUMBER,9X,6HLAST X,6X,15HLAST(
     11-M**M-KU))
C
C     BEGINNING OF CYCLE FOR NEW MACH NUMBER
    2 DIM=1.-DM*DM
      DK=DM**DM*2.4
      NN=NN+1
      IF(NN.GT.20) GO TO 500
C
C     START OF INTEGRATION PROCEDURE
      DBIGX=0.0
      DBIGU=-10.0
      DLASTX=0.0
      DLASTU=0.0
      DY(1)=0.0
      DDERY(1)=1.0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=(DPRMT(2)-DPRMT(1))/32.
      DPRMT(4)=1.E-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
C
C     LOGIC FOR CHOOSING NEW MACH NUMBER
      IF(IHLF.GT.10) GO TO 300
      IF(DLMDAE.LE.0.00) GO TO 300
      IF(DUDIFF.GT.1.E-6) GO TO 301
      GO TO 10
  300 DMUPBD=DM
```

```
202 FORMAT (1H ,3X,I3,4X,E12.5,5X,E12.5,5X,E12.5)
    WRITE (6,202) NN,DM,DLASTX,DLMDAE
    IF(DDM.LE.1.E-3) GO TO 10
    IF(DMLORD.LE.0.00) GO TO 210
    DDM=DDM/2.
210 DM=DM-DDM
    GO TO 2
301 DMLORD=DM
    WRITE (6,202) NN,DM,DLASTX,DLMDAE
    IF(DDM.LE.1.E-3) GO TO 10
    IF(DMUPBD.LE.0.00) GO TO 209
    DDM=DDM/2.
209 DM=DM+DDM
    GO TO 2
500 WRITE (6,501)
501 FORMAT (1H0,100HPROGRAM TERMINATED BECAUSE LOWER CRITICAL MACH NUM
   1BER NOT DETERMINED TO WITHIN .001 IN 20 ITERATIONS)
    GO TO 10
601 FORMAT (1H ,49HBODY OF REVOLUTION AND FLOW FIELD CHARACTERISTICS//
   1)
602 FORMAT (1H ,29HBODY MAX. THICKNESS AT X/L = ,3X,E12.5)
603 FORMAT (1H ,19HFINENESS RATIO F = ,13X,E12.5)
604 FORMAT (1H ,32HEXPONENT N FOR BODY ORDINATES = ,E12.5)
605 FORMAT (1H ,20HS--(X) = 0 AT X/L = ,12X,E12.5)
606 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,6X,E12.5)
303 WRITE (6,400)
400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
    GO TO 10
304 WRITE (6,401)
401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
    GO TO 10
306 WRITE (6,402)
402 FORMAT (113H FREE STREAM MACH NUMBER MUST BE GREATER THAN 0 AND LE
   1SS THAN THE LOWER CRITICAL MACH NUMBER WHICH IS LESS THAN 1 )
    GO TO 10
    END



    SUBROUTINE FCT(DX,DY,DZ)
    DIMENSION DY(1),DZ(1)
    COMMON /A/ D1M,DK
    COMMON /E/ DEF
    COMMON /R/ DLASTX,DLASTU,DLMDAE
    DLASTX=DX
    DA=DSPI(DX)
    DA2=DS2PI(DX)
    DA3=DS3PI(DX)
    DEF=DA2*ALOG(DA/(DX*(1.00-DX)))+DINT(DX)
    DLASTU=DY(1)+DEF
    DLMDAE=D1M-DK*DLASTU
    IF(DLMDAE.LE.0.00) RETURN
    DZ(1)=DA3*ALOG(DLMDAE)
    RETURN
    END



    SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
    DIMENSION DY(1),DDERY(1),DPRMT(5)
    COMMON /E/ DEF
    COMMON /S/ DBIGX,DBIGU,DUDIFF
    DU=DY(1)+DEF
    DUDIFF=DBIGU-DU
    IF(DUDIFF.GT.0.0) RETURN
    DBIGU=DU
    DBIGX=DX
    RETURN
    END
```

```
    FUNCTION DINT(DZ)
    EXTERNAL DFUN,DFUN1
    COMMON /G/ DW,DA2
    DW=DZ
    DA2=DS2PI(DZ)
    NIT=10
    DTOL=1.E-6
    CALL SIMP(0.00,DZ,NIT,NIT1,DTOL,DFUN,DANS)
    IF(NIT1.EQ.0) GO TO 20
    DINT1=DANS
    CALL SIMP(DZ,1.00,NIT,NIT1,DTOL,DFUN1,DANS)
    IF(NIT1.EQ.0) GO TO 20
    DINT2=DANS
    DINT=DINT1+DINT2
    RETURN
20  WRITE(6,21)
21  FORMAT (1H0,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
   1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
   2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X, 40HFOR THE GIVEN NUMBE
   3R OF ITERATIONS (NIT))
    WRITE(6,22) DZ
22  FORMAT(1H ,5X,4HX = ,E14.7)
    WRITE (6,23) DTOL
23  FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,E14.7)
    STOP
    END



    FUNCTION DFUN(DZ)
    COMMON /G/ DW,DA2
    IF (ABS(DZ-DW).LT.1.E-6) GO TO 20
    DB2=DS2PI(DZ)
    DFUN=(DA2-DB2)/(DW-DZ)
    RETURN
20  DFUN=DS3PI(DW)
    RETURN
    END



    FUNCTION DFUN1(DZ)
    COMMON /G/ DW,DA2
    IF (ABS(DZ-DW ).LT.1.E-6) GO TO 20
    DB2=DS2PI(DZ)
    DFUN1=(DA2-DB2)/(DZ-DW )
    RETURN
20  DFUN1=-DS3PI(DW)
    RETURN
    END



    FUNCTION DSPI(DZ)
    COMMON /C/ DCTAU2,DN
    COMMON /L/ K
    IF(K.GT.0) GO TO 2
    DS1=DZ-DZ**DN
 1  DSPI=DCTAU2*DS1*DS1/16.
    RETURN
 2  DS1=1.00-DZ-(1.00-DZ)**DN
    GO TO 1
    END
```

```
      FUNCTION DS2PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
    1 DS2PI=DCTAU2*(1.00-DN*DS1*(DN+1.00-(2.00*DN-1.00)*DS1))/8.00
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      GO TO 1
      END



      FUNCTION DS3PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(-DN-1.00+2.00*(2.00*DN-1.00)*DS1
     1*DZ))/8.00
      RETURN
    2 DS1=(1.00-DZ)**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(DN+1.00-2.00*(2.00*DN-1.00)*DS1*
     1(1.00-DZ)))/8.00
      RETURN
      END



      FUNCTION DRSQ(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
      DS2=1.00-DN*DS1
    1 DRSQ=DCTAU2*DS2*DS2/4.00
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      DS2=-1.00+DN*DS1
      GO TO 1
      END



      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      NIT1=10
      DH=(DXU-DXL)/2.00
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.00*DSUM2)/3.00
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.00
      DSUM3=0.00
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DK=K
    2 DSUM3=DSUM3+FUN(DXL+DH*DK)
      DANS=DH*(DSUM1+2.00*DSUM2+4.00*DSUM3)/3.00
      DERR=(DANS-DANS1)
      IF (ABS(DERR),LE.DTOL) RETURN
    1 DSUM2=DSUM2+DSUM3
      DTOL=DERR
      NIT1=0
      RETURN
      END
```

```
      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
      DIMENSION PRMT(1),Y(1),DERY(1),AUX(16,1)
      COMMON /B/ DLASTX,DLASTU,DLMDAE
      COMMON /S/ DBIGX,DBIGU,DUDIFF
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.0
      DO 1 I=1,NDIM
      AUX(16,I)=0.0
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C     ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C     COMPUTATION OF DERY FOR STARTING VALUES
    4 CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
C
C     RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(DUDIFF.GT.1.E-6) RETURN
      IF(PRMT(5))6,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C     COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.5*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(9,I)=DERY(I)
      ISW=3
      GO TO 100
C
C     COMPUTATION OF TEST VALUE DELT
   15 DELT=0.0
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.06666666667*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C     NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.
   18 IHLF=11
      X=X+H
```

```
      GO TO 4
C
C     THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
   22 Y(I)=AUX(1,I)+H*(.375  *AUX(8,I)+.79166667*AUX(9,I)-.20833333*AUX
     1(10,I)+.04166667*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(DUMDIFF.GT.1.E-6) RETURN
      IF(PRMT(5))6,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+.33333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+.375  *H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     ********************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+.4*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
      Z=X+.4*H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+.29697761*AUX(5,I)+.15875959*Z
C
      Z=X+.45573725*H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+.21810039*AUX(5,I)-3.05096515*AUX(6,I)+3.83286476*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 104 I=1,NDIM
  104 Y(I)=AUX(N,I)+.17476028*AUX(5,I)-.55148066*AUX(6,I)+1.2055356*AUX
     1(7,I)+.17118478*H*DERY(I)
      GO TO(9,13,15,21),ISW
C     ********************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
C
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
      DELT=AUX(N-4,I)+1.33333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+
     1AUX(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-.92561983*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
      DELT=.125  *(9.00*AUX(N-1,I)-AUX(N-3,I)+3.00*H*(DERY(I)+AUX(N+6,I)
     1+AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+.07438165*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0.0
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(DUMDIFF.GT.1.E-6) RETURN
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF (ABS(X-PRMT(2))-.100*ABS(H))212,215,215
  215 IF(DELT-.02*PRMT(4))216,216,201
C
```

```
C
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  221 AUX(16,I)=8.96296296*(Y(I)-AUX(N-3,I))-3.36111111*H*(DERY(I)+DELT
     1+AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=.5*H
      ISTEP=0
      DO 224 I=1,NDIM
      Y(I)=.390625E-2*(8.E1*AUX(N-1,I)+135.*AUX(N-2,I)+4.E1*AUX(N-3,I)+
     1AUX(N-4,I))-.1171875*(AUX(N+6,I)-6.*AUX(N+5,I)-AUX(N+4,I))*H
      AUX(N-4,I)=.390625E-2*(12.*AUX(N-1,I)+135.*AUX(N-2,I)+108.*AUX(N-3
     1,I)+AUX(N-4,I))-.0234375*(AUX(N+6,I)+18.*AUX(N+5,I)-9.*AUX(N+4,I))
     2*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  225 Y(I)=AUX(N-4,I)
      DELT=DELT-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      DO 226 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
      AUX(16,I)=8.96296296*(AUX(N-1,I)-Y(I))-3.36111111*H*(AUX(N+6,I)+
     1DELT+DERY(I))
  226 AUX(N+3,I)=DERY(I)
      GO TO 206
      END
```

```
C     PROGRAM FOR DETERMINING THE UPPER CRITICAL MACH NUMBER ON NONLIFTING
C     BODIES OF REVOLUTION HAVING ORDINATES R PROPORTIONAL TO X-X**N OR
C     1-X-(1-X)**N BY USING THE METHOD OF LOCAL LINEARIZATION--FOR REFERENCE
C     SEE SPREITER,J.R. AND ALKSNE,A.Y.,NASA TR-R2
C
C
C*********************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--INITIAL ESTIMATE OF UPPER CRITICAL MACH NUMBER
C     COLUMNS 11 TO 20--LOCATION AS FRACTION OF BODY LENGTH (X/L) OF
C     POSITION OF MAXIMUM BODY THICKNESS
C     COLUMNS 21 TO 30--FINENESS RATIO OF BODY
C*********************************************************************
C
C
C     MAIN PROGRAM
      REAL M
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION XMTKS(19),DNEXP(19),DZST(19)
      EXTERNAL FCT,OUTP
      COMMON /A/ D1M,DK
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      COMMON /R/ DLASTX,DLASTU,DLMDAH
      DATA XMTKS/0.05,.10,.15,.20,.25,.30,.35,.40,.45,.50,.55,.60,.65,
     1           .70,.75,.80,.85,.90,.95/
      DATA DNEXP/88.731,34.649,19.173,12.215,8.396,6.044,4.482,3.389,
     1           2.595,2.0,2.595,3.389,4.482,6.044,8.396,12.215,19.173,
     2           34.649,88.731/
      DATA DZST/0.0077126,0.019182,0.033578,0.050724,0.070564,0.093097,
     1          0.118356,0.146406,0.177347,0.211325,0.272310,0.337355,
     2          0.406294,0.479059,0.555659,0.636159,0.720673,0.809353,
     3          0.902386/
C
C     READING AND PRINTING OF INPUT DATA
   10 READ (5,200) M,XMT,F
  200 FORMAT (3F10.2)
      WRITE (6,600)
  600 FORMAT (1H1,35X78HCALCULATION OF UPPER CRITICAL MACH NUMBER FOR A
     1NONLIFTING BODY OF REVOLUTION /36X71HHAVING ORDINATES R PROPORTION
     2AL TO X-X**N OR 1-X-(1-X)**N BY USING THE /36X29HMETHOD OF LOCAL L
     3INEARIZATION ///)
      DM=M
      IF(XMT.GE.1..OR.XMT.LE.0.) GO TO 303
      IF(F.LT.0.) GO TO 304
      IF(M.LT.1.) GO TO 306
      K=1
      IF(XMT.GT..5) K=0
      DO 913 I=1,19
      J=I-1
      IF(XMTKS(I)-XMT) 913,914,951
  913 CONTINUE
  914 DN=DNEXP(I)
      DZS1=DZST(I)
      GO TO 950
  951 IF (I.EQ.1) GO TO 914
      DN=DNEXP(J)+(DNEXP(I)-DNEXP(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
      DZS1=DZST(J)+(DZST(I)-DZST(J))/(XMTKS(I)-XMTKS(J))*(XMT-XMTKS(J))
  950 CONTINUE
      A=XMT
      IF(XMT.LT..5) A=1.-XMT
      NIT=0
    5 DN1=DN-1.
      NIT=NIT+1
      IF(NIT.GT.20) GO TO 6
      AN1=A**DN1
      DNUM=1.-DN*AN1
      DENOM=-AN1*(DN*ALOG(A)+1.)
      DNEWN=DN-(DNUM/DENOM)
      ERR=ABS(DN/DNEWN-1.)
      DN=DNEWN
      IF(ERR.GT.1.E-4) GO TO 5
      GO TO 8
    6 WRITE (6,7)
    7 FORMAT (4X49HEXECUTION TERMINATED BECAUSE EXPONENT N CANNOT BE /5X
     149HDETERMINED TO WITHIN .01 PERCENT IN 20 ITERATIONS)
      GO TO 10
C
C     INITIALIZATION OF VARIOUS PARAMETERS
    8 DC=(DN**(DN/(DN-1.00)))/(DN-1.00)
      DMLOBD=-1.0
      DMUPBD=-1.0
      GAMMA=1.4
      NN=0
      DTAU=1./F
      DCTAU=DC*DTAU
      DCTAU2=DCTAU*DCTAU
      DZ0=0.0
      DZF=0.99
      DDM=0.10
C
C     CALCULATION OF THE POINT WHERE S--(X) = 0
      N=1
    1 DZS=DZS1-DS2PI(DZS1)/DS3PI(DZS1)
      IF(ABS(DZS-DZS1).LT.1.E-6) GO TO 40
      DZS1=DZS
      N=N+1
      IF(N.GT.10) GO TO 3
      GO TO 1
    3 WRITE (6,100)
  100 FORMAT (1H0,4X55HEXECUTION TERMINATED BECAUSE S--(X) = 0 POINT CAN
     1NOT BE /5X57HDETERMINED TO WITHIN SUFFICIENT ACCURACY IN 10 ITERAT
     2IONS)
      GO TO 10
   40 WRITE (6,601)
      WRITE (6,602) XMT
      WRITE (6,603) F
      WRITE (6,604) DN
      WRITE (6,605) DZS
      WRITE (6,606) GAMMA
      WRITE (6,99)
   99 FORMAT (1H0,//)
      WRITE (6,110)
  110 FORMAT (1H0,9HITERATION,2X,11HMACH NUMBER,9X,6HLAST X,6X,15HLAST (
     1M**M-1+KU))
C
C     BEGINNING OF CYCLE FOR NEW MACH NUMBER
    2 D1M=DM*DM-1.0
      DK=DM*DM*2.4
      NN=NN+1
      IF(NN.GT.20) GO TO 500
C
C     START OF INTEGRATION PROCEDURE
      DY(1)=0.0
      DDERY(1)=1.0
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
   24 DPRMT(3)=(DPRMT(2)-DPRMT(1))/32.
      DPRMT(4)=1.E-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
C
C     LOGIC FOR CHOOSING NEW MACH NUMBER
      IF(IHLF.GT.10) GO TO 301
      IF(DLMDAH.LE.0.0) GO TO 301
  300 DMUPBD=DM
      WRITE (6,202) NN,DM,DLASTX,DLMDAH
  202 FORMAT (1H ,3X,I3,4X,E12.5,5X,E12.5,5X,E12.5)
      IF(DDM.LE.1.E-3) GO TO 10
      IF(DMLOBD.LE.0.0) GO TO 210
      DDM=DDM/2.
  210 DM=DM-DDM
      GO TO 2
```

```
      DMLOBD=DM
      WRITE (6,202) NN,DM,DLASTX,DLMDAH
      IF(DDM.LE.1.E-3) GO TO 10
      IF(DMUPBD.LE.0.0) GO TO 209
      DDM=DDM/2.
  209 DM=DM+DDM
      GO TO 2
  500 WRITE (6,501)
  501 FORMAT (1H0,100HPROGRAM TERMINATED BECAUSE UPPER CRITICAL MACH NUM
     1BER NOT DETERMINED TO WITHIN .001 IN 20 ITERATIONS)
      GO TO 10
  601 FORMAT (1H ,49HBODY OF REVOLUTION AND FLOW FIELD CHARACTERISTICS//
     1)
  602 FORMAT (1H ,29HBODY MAX. THICKNESS AT X/L = ,3X,E12.5)
  603 FORMAT (1H ,19HFINENESS RATIO F = ,13X,E12.5)
  604 FORMAT (1H ,32HEXPONENT N FOR BODY ORDINATES = ,E12.5)
  605 FORMAT (1H ,20HS--(X) = 0 AT X/L = ,12X,E12.5)
  606 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,6X,E12.5)
  303 WRITE (6,400)
  400 FORMAT (1H0,42HXMT MUST BE GREATER THAN 0 AND LESS THAN 1)
      GO TO 10
  304 WRITE (6,401)
  401 FORMAT (1H0,42HFINENESS RATIO F MUST BE GREATER THAN ZERO)
      GO TO 10
  306 WRITE (6,402)
  402 FORMAT ( 88H FREE STREAM MACH NUMBER MUST BE GREATER THAN THE UPPE
     1R CRITICAL WHICH IS GREATER THAN 1)
      GO TO 10
      END



      SUBROUTINE FCT(DX,DY,DZ)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ D1M,DK
      COMMON /C/ DCTAU2,DN
      COMMON /E/ DHF
      COMMON /L/ K
      COMMON /R/ DLASTX,DLASTU,DLMDAH
      DLASTX=DX
      IF(DX.LT.1.E-6) GO TO 2
      DAA=DSPI(DX)/(DX*DX)
    1 DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      DHF=DA2*ALOG(DAA)+2.00*DINT(DX)
      DLASTU=DY(1)+DHF
      DLMDAH=D1M+DK*(DLASTU)
      IF(DLMDAH.LE.0.0) RETURN
      DZ(1)=DA3*ALOG(DLMDAH)
      RETURN
    2 IF(K.GT.0) GO TO 3
      DAA=DCTAU2/16.
      DX=0.
      GO TO 1
    3 DAA=DCTAU2*(DN-1.00)*(DN-1.00)/16.
      DX=0.
      GO TO 1
      END



      SUBROUTINE SIMP(DXL,DXU,NIT,NIT1,DTOL,FUN,DANS)
      NIT1=10
      DH=(DXU-DXL)/2.
      DSUM1=FUN(DXL)+FUN(DXU)
      DSUM2=FUN(DXL+DH)
      DANS=DH*(DSUM1+4.00*DSUM2)/3.
      N=2
      DO 1 I=1,NIT
      DANS1=DANS
      N=N*2
      DH=DH/2.
      DSUM3=0.0
      NLIM=N-1
      DO 2 K=1,NLIM,2
      DK=K
    2 DSUM3=DSUM3+FUN(DXL+DH*DK)
      DANS=DH*(DSUM1+2.00*DSUM2+4.00*DSUM3)/3.
      DERR=(DANS-DANS1)
      IF (ABS(DERR).LE.DTOL) RETURN
    1 DSUM2=DSUM2+DSUM3
      DTOL=DERR
      NIT1=0
      RETURN
      END



      FUNCTION DRSQ(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
      DS2=1.00-DN*DS1
    1 DRSQ=DCTAU2*DS2*DS2/4.
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      DS2=-1.00+DN*DS1
      GO TO 1
      END



      FUNCTION DS3PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(-DN-1.00+2.00*(2.00*DN-1.00)*DS1
     1*DZ))/8.
      RETURN
    2 DS1=(1.00-DZ)**(DN-2.00)
      DS3PI=(DCTAU2*DN*(DN-1.00)*DS1*(DN+1.00-2.00*(2.00*DN-1.00)*DS1*
     1(1.00-DZ)))/8.
      RETURN
      END



      FUNCTION DS2PI(DZ)
      COMMON /C/ DCTAU2,DN
      COMMON /L/ K
      IF(K.GT.0) GO TO 2
      DS1=DZ**(DN-1.00)
    1 DS2PI=DCTAU2*(1.00-DN*DS1*(DN+1.00-(2.00*DN-1.00)*DS1))/8.
      RETURN
    2 DS1=(1.00-DZ)**(DN-1.00)
      GO TO 1
      END
```

```
      FUNCTION DS2PI(DZ)
      COMMON /G/ DCTAU2,DN
      COMMON /Z/ K
      IF(K.GT.1) GO TO 2
      DS1=DZ-DZ**DN
    1 DS2PI=DCTAU2*DS1*DS1/16.
      RETURN
    2 DS1=1.00-DZ-(1.00-DZ)**DN
      GO TO 1
      END



      FUNCTION DFUN(DZ)
      COMMON /G/ DW,DA2
      IF (ABS(DZ-DW).LT.1.E-6) GO TO 20
      DB2=DS2PI(DZ)
      DFUN=(DA2-DB2)/(DW-DZ)
      RETURN
   20 DFUN=DS2PI(DW)
      RETURN
      END



      FUNCTION DINT(DZ)
      EXTERNAL DFUN
      COMMON /G/ DW,DA2
      IF (ABS(DZ).LT.1.E-6) GO TO 25
      DW=DZ
      DA2=DS2PI(DZ)
      NIT=10
      DTOL=1.E-6
      CALL SIMP(0.00,DZ,NIT,NIT1,DTOL,DFUN,DANS)
      IF(NIT1.EQ.0) GO TO 20
      DINT=DANS
      RETURN
   20 WRITE(6,21)
   21 FORMAT (1HO,2X,45HPROGRAM HAS TERMINATED BECAUSE THE FRACTIONAL/3X
     1, 47HERROR BETWEEN TWO SUCCESSIVE INTEGRALS OF DFUN /3X,45HIS LARG
     2ER THAN THE SPECIFIED TOLERANCE (DTOL)/3X, 40HOR THE GIVEN NUMBE
     3R OF ITERATIONS (NIT))
      WRITE (6,22) DZ
   22 FORMAT (1H ,5X,4HX = ,E14.7)
      WRITE (6,23) DTOL
   23 FORMAT (1H ,5X,29HFRACTIONAL ITERATIVE ERROR = ,3X,E14.7)
      STOP
   25 DINT=0.0
      RETURN
      END



      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,PRMT)
      DIMENSION DY(1),DDERY(1),PRMT(5)
      COMMON /F/ DHF
      RETURN
      END
```

```
      SUBROUTINE ODHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C
C
      DIMENSION PRMT(1),Y(1),DERY(1),AUX(16,1)
      COMMON /R/ DLASTX,DLASTU,DLMDAE
      N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.0
      DO 1 I=1,NDIM
      AUX(14,I)=0.0
      AUX(15,I)=DERY(I)
    1 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))3,2,4
C
C     ERROR RETURNS
    2 IHLF=12
      GO TO 4
    3 IHLF=13
C
C     COMPUTATION OF DERY FOR STARTING VALUES
    4 CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
C
C     RECORDING OF STARTING VALUES
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))16,5,6
    5 IF(IHLF)7,7,6
    6 RETURN
    7 DO 8 I=1,NDIM
    8 AUX(8,I)=DERY(I)
C
C     COMPUTATION OF AUX(2,I)
      ISW=1
      GO TO 100
C
    9 X=X+H
      DO 10 I=1,NDIM
   10 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTED BY MEANS OF BISECTION
   11 IHLF=IHLF+1
      X=X-H
      DO 12 I=1,NDIM
   12 AUX(4,I)=AUX(2,I)
      H=.5*H
      N=1
      ISW=2
      GO TO 100
C
   13 X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAE.LE.0.0) RETURN
      N=2
      DO 14 I=1,NDIM
      AUX(2,I)=Y(I)
   14 AUX(8,I)=DERY(I)
      ISW=3
      GO TO 100
C
C     COMPUTATION OF TEST VALUE DELT
   15 DELT=0.0
      DO 16 I=1,NDIM
   16 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.06666667*DELT
      IF(DELT-PRMT(4))19,19,17
   17 IF(IHLF-10)11,18,18
C
C     NO SATISFACTORY ACCURACY AFTER 10 BISECTIONS. ERROR MESSAGE.
   18 IHLF=11
      X=X+H
```

```
      GO TO 4
C
C     THERE IS SATISFACTORY ACCURACY AFTER LESS THAN 11 BISECTIONS.
   19 X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 20 I=1,NDIM
      AUX(3,I)=Y(I)
   20 AUX(10,I)=DERY(I)
      N=3
      ISW=4
      GO TO 100
C
   21 N=1
      X=X+H
      CALL FCT(X,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      X=PRMT(1)
      DO 22 I=1,NDIM
      AUX(11,I)=DERY(I)
   22 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.79166667*AUX(9,I)
     1-.20833333*AUX(10,I)+.041666667*DERY(I))
   23 X=X+H
      N=N+1
      CALL FCT(X,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))6,24,6
   24 IF(N-4)25,200,200
   25 DO 26 I=1,NDIM
      AUX(N,I)=Y(I)
   26 AUX(N+7,I)=DERY(I)
      IF(N-3)27,29,200
C
   27 DO 28 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
   28 Y(I)=AUX(1,I)+.33333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GO TO 23
C
   29 DO 30 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
   30 Y(I)=AUX(1,I)+.375*H*(AUX(8,I)+DELT+AUX(11,I))
      GO TO 23
C
C
C     ****************************************************************
C     THE FOLLOWING PART OF SUBROUTINE DHPCG COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD.
  100 DO 101 I=1,NDIM
      Z=H*AUX(N+7,I)
      AUX(5,I)=Z
  101 Y(I)=AUX(N,I)+.4*Z
C     Z IS AN AUXILIARY STORAGE LOCATION
C
      Z=X+.4*H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 102 I=1,NDIM
      Z=H*DERY(I)
      AUX(6,I)=Z
  102 Y(I)=AUX(N,I)+.29697761*AUX(5,I)+.15875964*Z
C
      Z=X+.45573725*H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 103 I=1,NDIM
      Z=H*DERY(I)
      AUX(7,I)=Z
  103 Y(I)=AUX(N,I)+.21810039*AUX(5,I)-3.05096515*AUX(6,I)+3.83286476*Z
C
      Z=X+H
      CALL FCT(Z,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 104 I=1,NDIM
  104 Y(I)=AUX(N,I)+.17476028*AUX(5,I)-.55148066*AUX(6,I)+1.2055356
     1*AUX(7,I)+.17118478*H*DERY(I)
      GO TO(9,13,15,21),ISW
C     ****************************************************************
C
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C
C     STARTING VALUES ARE COMPUTED.
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD.
  200 ISTEP=3
  201 IF(N-8)204,202,204
C
C     N=8 CAUSES THE ROWS OF AUX TO CHANGE THEIR STORAGE LOCATIONS
  202 DO 203 N=2,7
      DO 203 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  203 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSES N+1 TO GET N
  204 N=N+1
C
C     COMPUTATION OF NEXT VECTOR Y
      DO 205 I=1,NDIM
      AUX(N-1,I)=Y(I)
  205 AUX(N+6,I)=DERY(I)
      X=X+H
  206 ISTEP=ISTEP+1
      DO 207 I=1,NDIM
      DELT=AUX(N-4,I)+1.33333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+
     1AUX(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-.92561983*AUX(16,I)
  207 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILIARY STORAGE.
C
      CALL FCT(X,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
      DO 208 I=1,NDIM
      DELT=.125*(9.*AUX(N-1,I)-AUX(N-3,I)+3.*H*(DERY(I)+AUX(N+6,I)
     1+AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  208 Y(I)=DELT+.074380165*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0.0
      DO 209 I=1,NDIM
  209 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
      IF(DELT-PRMT(4))210,222,222
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  210 CALL FCT(X,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))212,211,212
  211 IF(IHLF-11)213,212,212
  212 RETURN
  213 IF(H*(X-PRMT(2)))214,212,212
  214 IF(ABS(X-PRMT(2))-.1D0*ABS(H))212,215,215
  215 IF(DELT-.02*PRMT(4))216,216,201
C
C
```

```
C     H COULD BE DOUBLED IF ALL NECESSARY PRECEDING VALUES ARE
C     AVAILABLE
  216 IF(IHLF)201,201,217
  217 IF(N-7)201,218,218
  218 IF(ISTEP-4)201,219,219
  219 IMOD=ISTEP/2
      IF(ISTEP-IMOD-IMOD)201,220,201
  220 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 221 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  221 AUX(16,I)=8.96296296*(Y(I)-AUX(N-3,I))-3.36111111*H*(DERY(I)+DELT
     1+AUX(N+4,I))
      GO TO 201
C
C
C     H MUST BE HALVED
  222 IHLF=IHLF+1
      IF(IHLF-10)223,223,210
  223 H=.5*H
      ISTEP=0
      DO 224 I=1,NDIM
      Y(I)=.390625E-2*(8.E1*AUX(N-1,I)+135.  *AUX(N-2,I)+4.E1*AUX(N-3,I)
     1+AUX(N-4,I))-.1171875  *(AUX(N+6,I)-6.  *AUX(N+5,I)-AUX(N+4,I))*H
      AUX(N-4,I)=.390625E-2*(12.  *AUX(N-1,I)+135.  *AUX(N-2,I)+
     1108.  *AUX(N-3,I)+AUX(N-4,I))-.0234375  *(AUX(N+6,I)+
     218.  *AUX(N+5,I)-9.  *AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  224 AUX(N+4,I)=AUX(N+5,I)
      X=X-H
      DELT=X-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 225 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  225 Y(I)=AUX(N-4,I)
      DELT=DELT-(H+H)
      CALL FCT(DELT,Y,DERY)
      IF(DLMDAH.LE.0.0) RETURN
      DO 226 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
      AUX(16,I)=8.96296296*(AUX(N-1,I)-Y(I))-3.36111111*H*(AUX(N+6,I)
     1+DELT+DERY(I))
  226 AUX(N+3,I)=DERY(I)
      GO TO 206
      END
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESSURE DISTRI-
C     BUTIONS FOR FREE STREAM MACH NUMBERS BELOW THE LOWER CRITICAL ON
C     NONLIFTING PARABOLIC-ARC BODIES, WHICH HAVE ELLIPTIC CROSS SECTIONS
C     THAT MAINTAIN A CONSTANT RATIO OF MAJOR TO MINOR AXES ALONG THE
C     ENTIRE LENGTH OF THE BODY, BY USING THE METHOD OF LOCAL LINEARIZA-
C     TION AND THE TRANSONIC EQUIVALENCE RULE--FOR REFERENCE SEE SPREITER,
C     J.R. AND ALKSNE,A.Y.,NASA TR-R2 AND HEASLET,M.A. AND SPREITER,J.R.,
C     NASA TR-1318
C
C
C*************************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGULAR LOCATIONS (IN DEGREES) ON THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE AND
C     AT MULTIPLES OF THE MAXIMUM DIAMETER D (= 1/FINENESS RATIO) OF THE
C     EQUIVALENT BODY OF REVOLUTION OUT TO A MAXIMUM DISTANCE OF 8D AND
C     AT THE FOUR INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE,
C     THETA, IN THE CROSS FLOW PLANE
C
C
C*************************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D),COMPLEX*16(C)
      COMPLEX*16 DCMPLX
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION CTH(4),C1(4),DTH(4),DTH1(4),DRB(4)
      EXTERNAL FCT,OUTP
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /CC/ DTAU2
      COMMON /X/ DL,DL1,DL2
      COMMON /Y/ C1,DTH1,DRB
      COMMON /MRMC/ DWRTE,DMRM
   10 READ (5,100) DM,DL,DF,DX,DTH
  100 FORMAT (8(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DM.GE.1..OR.DM.LE.0.) GO TO 306
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.14159265358979D0
      DGAMMA=1.400
      DMM=DX
      DZ0=.02D0
      DZF=.98D0
      DTAU=1.D0/DF
      DTAU2=DTAU*DTAU
      D1M=DM*DM-1.D0
      DK=DM*DM*2.4D0
      MM=1
      DZS=.211324865405187D0
      DO 500 I=1,4
      DTH1(I)=DTH(I)
      DTH(I)=DPI*DTH(I)/180.D0
      DAA=DCOS(DTH(I))
      DBB=DL*DSIN(DTH(I))
      DRB(I)=DL/DSQRT(DAA*DAA+DRB*DBB)
      CTH(I)=DCMPLX(0.D0,DTH(I))
  500 C1(I)=CDEXP(CTH(I))
      DL1=DL*DL-1.D0
      DL2=DSQRT(DL)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DGAMMA
      WRITE (6,606) DM
      DMRM=0.
      IF(DMM-.001) 920,920,921
  921 DWRTE=DZS/DMM+1.
      MRM=DWRTE
      DWRTE=MRM
      DWRTE=DWRTE*DMM+.001
      DMRM=-DMM
  920 CONTINUE
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO NOSE//)
      WRITE (6,5)
    5 FORMAT (5X,1HX,5X,10HTHETA(DEG),4X,8HCP(BODY),5X,6HCP(1D),6X,6HCP(
     12D),6X,6HCP(3D),6X,6HCP(4D),6X,6HCP(5D),6X,6HCP(6D),6X,6HCP(7D),6X
     2,6HCP(8D)/)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
      DPRMT(3)=-.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      WRITE (6,180)
   23 WRITE (6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO TAIL//)
      WRITE (6,5)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      GO TO 10
  600 FORMAT (1H1,35X67HCALCULATION OF THE SURFACE AND FLOW FIELD PRESSU
     1RE DISTRIBUTION FOR /36X65HPURELY SUBSONIC FLOW ABOUT A NONLIFTING
     2 PARABOLIC-ARC BODY, WHICH   /36X65HHAS AN ELLIPTIC CROSS SECTION
     3THAT MAINTAINS A CONSTANT RATIO OF  /36X65HMAJOR TO MINOR AXES ALO
     4NG THE ENTIRE LENGTH OF THE BODY, BY USING /36X63HTHE METHOD OF LO
     5CAL LINEARIZATION AND THE TRANSONIC EQUIVALENCE /36X4HRULE///)
  601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
  602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
  603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
  604 FORMAT (1H ,20HS''(X) = 0 AT X/L = ,16X,D12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
  606 FORMAT (1H ,26HFREE STREAM MACH NUMBER = ,10X,D12.5)
  304 WRITE (6,400)
  400 FORMAT (1H0,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
      GO TO 10
```

```
305 WRITE (6,401)
401 FORMAT (1H0,54HRATIO OF MAJOR TO MINOR AXES  MUST BE GREATER THAN 1 (
   1ERO)
    GO TO 10
306 WRITE (6,402)
402 FORMAT (113H FREE STREAM MACH NUMBER MUST BE GREATER THAN 0 AND LE
   1SS THAN THE LOWER CRITICAL MACH NUMBER WHICH IS LESS THAN 1)
    GO TO 10
307 WRITE (6,403)
403 FORMAT (1H0,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION OUTPUT-UNIT
   1MUST BE GREATER THAN 0 AND LESS THAN 1 )
    GO TO 10
    END



    SUBROUTINE FCT(DX,DY,DZ)
    IMPLICIT REAL*8(D)
    DIMENSION DY(1),DZ(1)
    COMMON /A/ DIM,DK
    COMMON /B/ MM
    COMMON /E/ DEF
    DA=DSPI(DX)
    DA2=DS2PI(DX)
    DA3=DS3PI(DX)
    DEF=DA2*(DLOG(DA/(DX*(1.D0-DX)))+3.D0)
    DUU=-DIM-DK*(DY(1)+DEF)
    IF(DUU) 2,2,1
  1 DZ(1)=DA3*DLOG(DUU)
    RETURN
  2 MM=2
    WRITE (6,700) DX
700 FORMAT (1H0,43HLOG ARGUMENT (1-DSM-KU) IS NEGATIVE AT X = ,D12.5)
    WRITE (6,701)
701 FORMAT (1H ,72HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER GREATE
   1R THAN LOWER CRITICAL)
    RETURN
    END



    SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
    IMPLICIT REAL*8(D),COMPLEX*16(C)
    DIMENSION DY(1),DDERY(1),DPRMT(5),DCPF(8),DTH1(4),C1(4),DRB(4)
    COMMON /E/ DEF
    COMMON /MRMC/ DWRTE,DMRM
    COMMON /X/ DL,DL1,DL2
    COMMON /Y/ C1,DTH1,DRB
    IF(IHLF.GT.10) GO TO 10
    IF(DX.LE..01.OR.DX.GE..99) GO TO 30
    IF(DMRM) 50,30,60
 50 IF(DX.GT.DWRTE) RETURN
    GO TO 30
 60 IF(DX.LT.DWRTE) RETURN
 30 DU=DY(1)+DEF
    DCP1=2.D0*DU
    DA=2.D0*DS2PI(DX)
    DB=DRSO(DX)
    DRF=.5D0*DL2/(DX*(1.D0-DX))
    DO 1 I=1,4
    CR=DRB(I)*C1(I)
    CALL CPFLD(CR,DF1,DF2)
    DCPB=-DA*DF1+DB*DF2-DCP1
    DO 2 J=1,8
    DJ=J
    CR=DRF*DJ*C1(I)
    CALL CPFLD(CR,DF1,DF2)
  2 DCPF(J)=-DA*DF1+DB*DF2-DCP1
  1 WRITE (6,100) DX,DTH1(I),DCP1,DCPF
100 FORMAT (1H ,F6.5,1X1PD11.4,1X,6(1X,1PD11.4))
    WRITE (6,101)
101 FORMAT (1H0)
    DWRTE=DWRTE+DMRM
    RETURN
 10 WRITE (6,102)
102 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
   1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
   2ZE (.001) 10 TIMES )
    GO TO 30
    END



    SUBROUTINE CPFLD(CR,DF1,DF2)
    IMPLICIT REAL*8(D),COMPLEX*16(C)
    COMPLEX*16 DCONJG,CDSQRT
    COMMON /X/ DL,DL1,DL2
    CRSQ=CR*CR
    CR1=CRSQ-DL1
    CR2=CDSQRT(CR1)
    CR3=DCONJG(CR2)
    CR4=CR+CR2
    CR5=DCONJG(CR4)
    CR6=1.D0/(CR2*CR4)
    CR7=DCONJG(CR6)
    D1=CR4*CR5
    DF1=DLOG(D1)/(4.D0*DL)
    D2=DL1*(CR6+CR7)
    D3=CR2*CR3
    D4=DL/D3
    DF2=D2-D4
    RETURN
    END



    FUNCTION DRSO(DZ)
    IMPLICIT REAL*8(D)
    COMMON /CC/ DTAU2
    DS1=1.D0-2.D0*DZ
    DRSO=4.D0*DTAU2*DS1*DS1
    RETURN
    END



    FUNCTION DSPI(DZ)
    IMPLICIT REAL*8(D)
    COMMON /CC/ DTAU2
    DS1=DZ*(1.D0-DZ)
    DSPI=DTAU2*DS1*DS1
    RETURN
    END



    FUNCTION DS2PI(DZ)
    IMPLICIT REAL*8(D)
    COMMON /CC/ DTAU2
    DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
    RETURN
    END
```

```
      FUNCTION DS3P1(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3P1=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END


      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESSURE DISTRI-
C     BUTIONS FOR FREE STREAM MACH NUMBERS ABOVE THE UPPER CRITICAL ON
C     NONLIFTING PARABOLIC-ARC BODIES, WHICH HAVE ELLIPTIC CROSS SECTIONS
C     THAT MAINTAIN A CONSTANT RATIO OF MAJOR TO MINOR AXES ALONG THE
C     ENTIRE LENGTH OF THE BODY, BY USING THE METHOD OF LOCAL LINEARIZA-
C     TION AND THE TRANSONIC EQUIVALENCE RULE--FOR REFERENCE SEE SPREITER,
C     J.R. AND ALKSNE,A.Y.,NASA TR-R2 AND HEASLET,M.A. AND SPREITER,J.R.,
C     NASA TR-1318
C
C
C*****************************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGULAR LOCATIONS (IN DEGREES) ON THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE AND
C     AT MULTIPLES OF THE MAXIMUM DIAMETER D (= 1/FINENESS RATIO) OF THE
C     EQUIVALENT BODY OF REVOLUTION OUT TO A MAXIMUM DISTANCE OF 8D AND
C     AT THE FOUR INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE,
C     THETA, IN THE CROSS FLOW PLANE
C
C
C*****************************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D),COMPLEX*16(C)
      COMPLEX*16 DCMPLX
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION CTH(4),C1(4),DTH(4),DTH1(4),DRB(4)
      EXTERNAL FCT,OUTP
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /CC/ DTAU2
      COMMON /F/ DZS
      COMMON /X/ DL,DL1,DL2
      COMMON /Y/ C1,DTH1,DRB
      COMMON /MRMC/ DWRTE,DMRM
   10 READ (5,100) DM,DL,DF,DX,DTH
  100 FORMAT (8(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DM.LE.1.) GO TO 306
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.141592653589793D0
      DGAMMA=1.4D0
      DMM=DX
      DZO=.02D0
      DZF=.98D0
      DTAU=1.D0/DF
      DTAU2=DTAU*DTAU
      D1M=DM*DM-1.D0
      DK=DM*DM*2.4D0
      MM=1
      DZS=.211324865405187ID0
      DO 500 I=1,4
      DTH1(I)=DTH(I)
      DTH(I)=DPI*DTH(I)/180.D0
      DAA=DCOS(DTH(I))
      DBB=DL*DSIN(DTH(I))
      DRB(I)=DL/DSQRT(DAA*DAA+DBB*DBB)
      CTH(I)=DCMPLX(0.D0,DTH(I))
  500 C1(I)=CDEXP(CTH(I))
      DL1=DL*DL-1.D0
      DL2=DSQRT(DL)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DGAMMA
      WRITE (6,606) DM
      DMRM=0.
      IF(DMM-1.001) 920,920,921
  921 DWRTE=DZS/DMM+1.
      MRM=DWRTE
      DWRTE=MRM
      DWRTE=DWRTE*DMM+.001
      DMRM=-DMM
  920 CONTINUE
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO NOSE//)
      WRITE (6,5)
    5 FORMAT (5X,1HX,5X,10HTHETA(DEG),4X,8HCP(BODY),5X,6HCP(1D),6X,6HCP(
     12D),6X,6HCP(3D),6X,6HCP(4D),6X,6HCP(5D),6X,6HCP(6D),6X,6HCP(7D),6X
     2,6HCP(8D)/)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZO
      DPRMT(3)=-.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      WRITE (6,180)
   23 WRITE (6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO TAIL//)
      WRITE (6,5)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      GO TO 10
  600 FORMAT (1H1,35X67HCALCULATION OF THE SURFACE AND FLOW FIELD PRESSU
     1RE DISTRIBUTION FOR /36X67HPURELY SUPERSONIC FLOW ABOUT A NONLIFTI
     2NG PARABOLIC-ARC BODY, WHICH /36X65HHAS AN ELLIPTIC CROSS SECTION
     3THAT MAINTAINS A CONSTANT RATIO OF  /36X65HMAJOR TO MINOR AXES ALO
     4NG THE ENTIRE LENGTH OF THE BODY, BY USING /36X63HTHE METHOD OF LO
     5CAL LINEARIZATION AND THE TRANSONIC EQUIVALENCE /36X4HRULE///)
  601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
  602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
  603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
  604 FORMAT (1H ,20HS''(X) = 0 AT X/L = ,16X,D12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
  606 FORMAT (1H ,26HFREE STREAM MACH NUMBER = ,10X,D12.5)
  304 WRITE (6,400)
  400 FORMAT (1H0,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
```

```
      GO TO 10
  305 WRITE (6,401)
  401 FORMAT (1H0,54HRATIO OF MAJOR TO MINOR AXES MUST BE GREATER THAN Z
     1ERO)
      GO TO 10
  306 WRITE (6,402)
  402 FORMAT (1H0,100HFREE STREAM MACH NUMBER MUST BE GREATER THAN THE U
     1PPER CRITICAL MACH NUMBER WHICH IS GREATER THAN 1 )
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (1H0,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT
     1MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END
```

```
      SUBROUTINE FCT(DX,DY,DZ)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ DIM,DK
      COMMON /B/ MM
      COMMON /E/ DHF
      COMMON /F/ DZS
      DA=DSPI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      DHF=DA2*DLOG(DA/(DX*DX))+2.D0*DINT(DX)
      DUU=DIM+DK*(DY(1)+DHF)
      IF(DUU) 2,2,1
    1 DZ(1)=DA3*DLOG(DUU)
      RETURN
    2 MM=2
      WRITE (6,700) DX
  700 FORMAT (1H0,43HLOG ARGUMENT (M*M-1+KU) IS NEGATIVE AT X = ,D12.5)
      IF(DX-DZS) 3,4,4
    3 WRITE (6,701)
  701 FORMAT (1H ,69HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER LESS T
     1HAN UPPER CRITICAL )
    4 RETURN
      END
```

```
      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
      IMPLICIT REAL*8(D),COMPLEX*16(C)
      DIMENSION DY(1),DDERY(1),DPRMT(5),DCPF(8),DTH1(4),C1(4),DRB(4)
      COMMON /E/ DHF
      COMMON /MRMC/ DWRTE,DMRM
      COMMON /X/ DL,DL1,DL2
      COMMON /Y/ C1,DTH1,DRB
      IF(IHLF.GT.10) GO TO 10
      IF(DX.LE..01.OR.DX.GE..99) GO TO 30
      IF(DMRM) 50,30,60
   50 IF(DX.GT.DWRTE) RETURN
      GO TO 30
   60 IF(DX.LT.DWRTE) RETURN
   30 DU=DY(1)+DHF
      DCP1=2.D0*DU
      DA=2.D0*DS2PI(DX)
      DR=DRSO(DX)
      DRF=.5D0*DL2/(DX*(1.D0-DX))
      DO 1 I=1,4
      CR=DRB(I)*C1(I)
      CALL CPFLD(CR,DF1,DF2)
      DCPR=-DA*DF1+DB*DF2-DCP1
      DO 2 J=1,8
      DJ=J
      CR=DRF*DJ*C1(I)
      CALL CPFLD(CR,DF1,DF2)
    2 DCPF(J)=-DA*DF1+DR*DF2-DCP1
    1 WRITE (6,100) DX,DTH1(I),DCP8,DCPF
  100 FORMAT (1H ,F8.5,1X1PD11.4,1X,9(1X,1PD11.4))
      WRITE (6,101)
  101 FORMAT (1H0)
      DWRTE=DWRTE+DMRM
      RETURN
   10 WRITE (6,102)
  102 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      GO TO 30
      END
```

```
      SUBROUTINE CPFLD(CR,DF1,DF2)
      IMPLICIT REAL*8(D),COMPLEX*16(C)
      COMPLEX*16 DCONJG,CDSQRT
      COMMON /X/ DL,DL1,DL2
      CRSQ=CR*CR
      CR1=CRSQ-DL1
      CR2=CDSQRT(CR1)
      CR3=DCONJG(CR2)
      CR4=CR+CR2
      CR5=DCONJG(CR4)
      CR6=1.D0/(CR2*CR4)
      CR7=DCONJG(CR6)
      D1=CR4*CR5
      DF1=DLOG(D1/(4.D0*DL))
      D2=DL1*(CR6+CR7)
      D3=CR2*CR3
      D4=DL/D3
      DF2=D2-D4
      RETURN
      END
```

```
      FUNCTION DINT(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DINT=12.D0*DTAU2*DZ*(-1.D0+1.5D0*DZ)
      RETURN
      END
```

```
      FUNCTION DRSQ(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=1.D0-2.D0*DZ
      DRSQ=4.D0*DTAU2*DS1*DS1
      RETURN
      END
```

```
      FUNCTION DSPI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=DZ*(1.D0-DZ)
      DSPI=DTAU2*DS1*DS1
      RETURN
      END
```

```
      FUNCTION DS2PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
      RETURN
      END



      FUNCTION DS3PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3PI=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END



      SUBROUTINE DHPCG(PRMT,Y,DERF,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1.0 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE AND FLOW FIELD PRESS... FIST I-
C     BUTIONS FOR FREE STREAM MACH NUMBERS AT ZERO ANGLE OF ATTACK
C     PARABOLIC-ARC BODIES, WHICH HAVE ELLIPTIC CROSS SECTIONS AND MAI-
C     TAIN A CONSTANT RATIO OF MAJOR TO MINOR AXES ALONG THE B TIRE
C     LENGTH OF THE BODY, BY USING THE METHOD OF LOCAL LINEARIZATION AND
C     THE TRANSONIC EQUIVALENCE RULE--FOR DETAILS OF SEE SPECIFICATIONS
C     AND ALKSNE,A.Y.,NASA TR-R2 AND PEASLEY, ... NASA TR-1318
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS ... IS-
C     PUTTED ON ONE CARD AS FOLLOWS
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGULAR LOCATIONS (IN DEGREES)    THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE A T
C     AT MULTIPLES OF THE MAXIMUM DIAMETER D (= 1/FINENESS RATIO) OF THE
C     EQUIVALENT BODY OF REVOLUTION OUT TO A RADIAL DISTANCE ... AND A
C     AT THE FOUR INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE,
C     THETA, IN THE CROSS FLOW PLANE
C
C
C*****************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D),COMPLEX*16(C)
      COMPLEX*16 DCMPLX
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION CTH(4),C1(4),DTH(4),DTH1(4),DFF(4)
      EXTERNAL FCT,OUTP
      COMMON /A/ DM,DK,D1,D1M,DZS,DU1,DU2,DU3,D...,DU5,
      COMMON /E/ DX11,DX12,DU01,DU02
      COMMON /CC/ DTAU2
      COMMON /B/ MM
      COMMON /F/ DV,DV3
      COMMON /X/ DL,DL1,DL2
      COMMON /Y/ C1,DTH1,DRB
      COMMON /MRMC/ DWRTE,DMRM
   10 READ (5,100) DM,DL,DF,DX,DTH
  100 FORMAT (8(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.1415926535897900
      DGAMMA=1.400
      DMM=DX
      DZO=.0200
      DZF=.9800
      DTAU=1.D0/DF
      DTAU2=DTAU*DTAU
      D1M=DM*DM-1.D0
      DK=DM*DM*2.400
      D1=DK*1.7810724179901900
      M=1
      MM=1
      DZS=.21132486540518710
      DX12=0.D0
      DU02=0.D0
      DO 500 I=1,4
      DTH1(I)=DTH(I)
      DTH(I)=DPI*DTH(I)/180.D0
      DAM=DCOS(DTH(I))
      DSM=DSIN(DTH(I))
      C(I)=DL*DSQRT(DAM*DAM+DF*DF*DSM*DSM)*
     1DTH(I)=DF*DATAN(D0,DTH(I))
  500 C1(I)=C(I)/(2.*DTH(I))
      DL1=DL*DL-1.D0
      DL2=DSQRT(DL)
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DCAREA
      WRITE (6,606) DV
      DWRTE=0.
      IF(DF-.001) 920,920,921
  921 DWRTE=DZS/DFM+1.
      DMM=DWRTE
      DWRTE=MMM
      DWRTE=DWRTE*DMM+.001
      DMRM=-DMM
  920 CONTINUE
C
C     CALCULATION OF AREA AND DERIVATIVES OF AREA AT XS
C
      DA=DSPI(DZS)
      DA1=DS1PI(DZS)
      DA2=DS2PI(DZS)
      DA3=DS3PI(DZS)
      DA4=DS4PI(DZS)
      DA5=DS5PI(DZS)
      DA6=DS6PI(DZS)
      DA7=DS7PI(DZS)
      DAM=DLOG(D1*DA/DZS)
      D1N=DA1/DA
      D3N=DA3/DA
      D4N=DA4/DA
      D1N2=D1N*D1N
      D1N3=D1N*D1N2
C
C     CALCULATION OF INITIAL VALUE OF U
C
      DU=-D1M*D1N*T(DZS)
C
C     CALCULATION OF THE 1ST DERIVATIVE OF U
C
      DU1=12.D0*DTAU2*(-1.D0+3.D0*DZS)
      DU1=5.D-1
      N=1
   13 DU11=DU1-DU1*(DU1-DA3*(DLOG(DU1)+DAM)-D1T)/(DU1-DA3)
      IF(DABS(DU1)-DU11).LT.1.D-10) GO TO 14
      DU1=DU11
      N=N+1
      IF(N.GT.11) GO TO 200
      GO TO 13
  200 WRITE (6,104)
  104 FORMAT (1H0,4X73HPROGRAM TERMINATED BECAUSE INITIAL DERIVATIVE OF
     1U AT S**(X) = 0 REQUIRED /5X48HFOR TAYLOR SERIES START AT THAT POI
     2NT CANNOT BE DETERMINED TO WITHIN /5X36HSUFFICIENT ACCURACY IN 10
     3ITERATIONS)
      GO TO 10
   14 DU1=DU11
C
C     CALCULATION OF THE INITIAL 2ND DERIVATIVE OF U
C
      DZ1=36.D0*DTAU2
      DA3=DA3/DU1
      DA4=DA4/DU1**2
      DU2=(DA4+DA3*(1+2.D0*DA3*(D1*(1.D0-DA3)-1.D0/DZS)+D21)/(1.D0-2.D0
     1*DA3)
C
C     CALCULATION OF THE INITIAL 3RD DERIVATIVE OF U
```

```
      D3I=0.D0
      D21=DU2/DU1-DZ3I*D1A
      D2C=D21+D1N-1.D0/DZS
      DA41=DA4/DU1
      D3II=D1N*(DA41-2.D0*DA3I*D1N)
      D3C1=-D21*D21-D1N2+1.D0/(DZS*D/S)
      DU3=(DA5*DAM1+3.D0*DA4*D2C+3.D0*D3I*(D3C1-D3II)+D3I)/(1.D0-3.D0*
     1DA3I)
C
C     CALCULATION OF THE INITIAL 4TH DERIVATIVE OF U
C
      D4I=0.D0
      D3I=DU3/DU1-D3II
      D3C=D3I+D3C1
      D4II=(D1N*(DA5-3.D0*DA4*D1N)+D3I*(6.D0*DA1*D1N2+DA3I))/DU1
      D4C1=-3.D0*D3I*D21*D21+2.D0*(D21**3)+D3I*2.D0*D1N2-2.D0/(DZS**3)
      DU4=(DA6*DA1+4.D0*DA5*D2C+6.D0*DA4*D3C+4.D0*DA3I*(-D4II+D4C1)+D4I
     1I)/(1.D0-4.D0*DA3I)
C
C     CALCULATION OF THE INITIAL 5TH DERIVATIVE OF U
C
      D5I=0.D0
      D5II=(D1N*(DA6-4.D0*DA5*D1N)+4.D0*DA4*D1N**3+(D3I*2.D0*D1N2)*(4.D0
     1*DA4-12.D0*DA3*D1N))/DU1
      D41=DU4/DU1-D4II
      D4C=D41+D4C1
      DU5=(DA7*DAM1+5.D0*DA6*D2C+10.D0*DA5*D3C+10.D0*DA4*D4C+5.D0*DA3*
     1(-D5II-4.D0*D41*D21+12.D0*D3I*D21*D21-3.D0*D3I*D3I-6.D0*(D21**4)+
     2D4N-4.D0*D3N*D1N-6.D0*D1N2+6.D0/(DZS**4))+D5I)/(1.D0-5.D0*DA3I)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,101)
  101 FORMAT (1HO,44HSTART OF INTEGRATION FROM S**(X) = 0 TO NOSE//)
      WRITE (6,5)
    5 FORMAT (5X,1HX,5X,10HTHETA(DEG),4X,8HCP(BODY),5X,6HCP(1D),6X,6HCP(
     12D),6X,6HCP(3D),6X,6HCP(4D),6X,6HCP(5D),6X,6HCP(6D),6X,6HCP(7D),6X
     2,6HCP(8D)/)
      DY(1)=DU
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
      DPRMT(3)=-.00100
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      WRITE (6,180)
   23 WRITE (6,103)
  103 FORMAT (1HO,44HSTART OF INTEGRATION FROM S**(X) = 0 TO TAIL//)
      WRITE (6,5)
      DPRMT(1)=DZS
      DPRMT(2)=DZE
      DY(1)=DU
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=WRM
      DWRTE=DWRTE*DM3
      DMRM=DMR
  922 CONTINUE
   27 DPRMT(3)=.00100
      DPRMT(4)=1.D-6
      DDERY(1)=1.D0
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.EQ.2) GO TO 29
      IF(MM.EQ.3) GO TO 30
   28 GO TO 10
   29 DY(1)=DV
      DPRMT(1)=DV3
      MM=1
      WRITE (6,199)
  199 FORMAT (1HO,31HSTART OF SUBSONIC CALCULATION)
      WRITE (6,198) DV3
  198 FORMAT (1HO,37HSUBSONIC CALCULATION STARTS AT X = ,D12.5//)
      WRITE (6,5)
      GO TO 27
   30 DSL=(DX12-DX11)/(DUU1-DUU2)
      DSH=DSL*(DUU2+D14)
      DX3=DX12+DSH
      DX4=2.D0*DX3-DX12
      WRITE (6,205)
  205 FORMAT (1HO,29HSTART OF SUBSONIC CALCULATION)
      WRITE (6,509) DX3
  509 FORMAT (1HO,35HLOG ARGUMENT (R*M-1+KU) = 0 AT X = ,D12.5)
      WRITE (6,510) DX4
  510 FORMAT (1HO,35HSUBSONIC CALCULATION BEGINS AT X = ,D12.5//)
      DY(1)=-DUU2-2.D0*D1N
      DPRMT(1)=DX4
      MM=1
      WRITE (6,5)
      GO TO 27
  600 FORMAT (1H1,35X67HCALCULATION OF THE SURFACE AND FLOW FIELD PRESSU
     1RE DISTRIBUTION FOR /36X67HFREE STREAM MACH NUMBERS AT OR NEAR 1 A
     2BOUT A MINLIFTING PARABOLIC- /36X62HARC BODY, WHICH HAS AN ELLIPTI
     3C CROSS SECTION THAT MAINTAINS A /36X63HCONSTANT RATIO OF MAJOR TO
     4MINOR AXES ALONG THE ENTIRE LENGTH OF /36X60HTHE BODY, BY USING TH
     5E METHOD OF LOCAL LINEARIZATION AND THE /36X26HTRANSONIC EQUIVALEN
     6CE RULE///)
  601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
  602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
  603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
  604 FORMAT (1H ,20HS**(2) = 0 AT X/L = ,16X,D12.5)
  605 FORMAT (1H ,24HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
  606 FORMAT (1H ,24HFREE STREAM MACH NUMBER = ,10X,D12.5)
  304 WRITE (6,400)
  400 FORMAT (1HO,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
      GO TO 10
  305 WRITE (6,401)
  401 FORMAT (1HO,54HRATIO OF MAJOR TO MINOR AXES MUST BE GREATER THAN Z
     1ERO)
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (1HO,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT
     1MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END



      SUBROUTINE FCT (DX,DY,DZ)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ PI,DR,T1,T2,DZS,DU1,DU2,DU3,DU4,DU5,
      COMMON /I/ MM
      COMMON /E/ DV,DV3
      IF(DABS(DX-DZS).LE.5.D-3) GO TO 10
      DZ=DSPI(DX)
      DA1=DS1PI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      IF(MM.EQ.1) GO TO 1
      IF(MM.EQ.2) GO TO 2
      IF(MM.EQ.3) GO TO 3
    1 IF(DX-DZS) 20,20,30
```

```
20 DZ(1)=DA1*DA2/DA+DEXP((DY(1)+DL[illegible]-D[illegible]T(DX)-[illegible] DLOG(D[illegible])/[illegible])
   RETURN
30 DUU=DM*DM-1.D0+DK*DY(1)
   IF(DUU) 20,20,40
40 DZH=DA3*DLOG(DUU*DA/(DX*DX))+DA2*(DA1/DA-2.D0/DX)+2.D0[illegible]
   DZ(1)=DA1*DA2/DA+DEXP((DY(1)+D1M-DINT(DX)-DA2*DLOG(D[illegible])/DA2)
   DIFF=DZH-DZ(1)
   IF(DIFF) 60,60,50
50 RETURN
60 M=2
   MM=2
   DV=DY(1)
   DV3=DX
   RETURN
 2 DUU=DM*DM-1.D0+DK*DY(1)
   IF(DUU) 80,80,70
70 DZ(1)=DA3*DLOG(DUU*DA/(DX*DX))+DA2*(DA1/DA-2.D0/DX)+2.D0*D1I[illegible]T(DX)
   RETURN
80 M=3
   MM=3
   RETURN
 3 DUU=1.D0-DM*DM-DK*DY(1)
 5 DZ(1)=DA3*DLOG(DUU*DA/(DX*(1.D0-DX)))+DA2*(DA1/DA-1.D0/DX+1.D0
  1/(1.D0-DX))+D1INT(DX)+D1INT1(DX)
   RETURN
10 DZ(1)=DU1+DU2*(DX-DZS)+DU3*((DX-DZS)**2)/2.D0+DU4*((DX-DZS)**3)/
  16.D0+DU5*((DX-DZS)**4)/24.D0
   RETURN
   END


   SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
   IMPLICIT REAL*8(D),COMPLEX*16(C)
   DIMENSION DY(1),DDERY(1),DPRMT(5),DCPF(8),DTH1(4),C1(4),DRB(4)
   COMMON /E/ DX11,DX12,DUU1,DUU2
   COMMON /X/ DL,DL1,DL2
   COMMON /Y/ C1,DTH1,DRB
   COMMON /MRMC/ DWRTE,DMRM
   DX11=DX12
   DX12=DX
   DUU1=DUU2
   DUU2=DY(1)
   IF(IHLF.GT.10) GO TO 10
   IF(DX.LE..01.OR.DX.GE..99) GO TO 30
   IF(DMRM) 50,30,60
50 IF(DX.GT.DWRTE) RETURN
   GO TO 30
60 IF(DX.LT.DWRTE) RETURN
30 DCP1=2.D0*DY(1)
   DA=2.D0*DS2PI(DX)
   DB=DRSO(DX)
   DRF=.5D0*DL2/(DX*(1.D0-DX))
   DO 1 I=1,4
   CR=DRB(I)*C1(I)
   CALL CPFLD(CR,DF1,DF2)
   DCPB=-DA*DF1+DB*DF2-DCP1
   DO 2 J=1,8
   DJ=J
   CR=DRF*DJ*C1(I)
   CALL CPFLD(CR,DF1,DF2)
 2 DCPF(J)=-DA*DF1+DB*DF2-DCP1
 1 WRITE (6,100) DX,DTH1(I),DCPB,DCPF
100 FORMAT (1H ,F8.5,1X1PD11.4,1X,9(1X,1PD11.4))
   WRITE (6,101)
101 FORMAT (1H0)
   DWRTE=DWRTE+DMRM
   RETURN
10 WRITE (6,102)
102 F[illegible] BECAUSE [illegible]
  1[illegible] ITERATION [illegible] TO [illegible]
  2/F [illegible] TI[illegible]S )
   GO TO 30
   END


   SUBROUTINE CPFLD(CR,DF1,DF2)
   IMPLICIT REAL*8(D),COMPLEX*16(C)
   COMPLEX*16 DCONJG,CDSQRT
   COMMON /X/ DL,DL1,DL2
   CRSQ=CR*CR
   CR1=CRSQ-DL1
   CR2=CDSQRT(CR1)
   CR3=DCONJG(CR2)
   CR4=CR+CR2
   CR5=DCONJG(CR4)
   CR6=1.D0/(CR2*CR4)
   CR7=DCONJG(CR6)
   D1=CR4*CR5
   DF1=DLOG(D1/(4.D0*DL))
   D2=DL1*(CR6+CR7)
   D3=CR2*CR3
   D4=DL/D3
   DF2=D2-D4
   RETURN
   END


   FUNCTION DINT(DZ)
   IMPLICIT REAL*8(D)
   COMMON /CC/ DTAU2
   DINT=12.D0*DTAU2*DZ*(-1.D0+1.5D0*DZ)
   RETURN
   END


   FUNCTION DINT1(DX)
   IMPLICIT REAL*8(D)
   COMMON /CC/ DTAU2
   DINT1=12.D0*DTAU2*(.5D0-2.D0*DX+1.5D0*DX*DX)
   RETURN
   END


   FUNCTION D1INT(DX)
   IMPLICIT REAL*8(D)
   COMMON /CC/ DTAU2
   D1INT=12.D0*DTAU2*(-1.D0+3.D0*DX)
   RETURN
   END


   FU[illegible]T[illegible]I[illegible]T1(DX)
   I[illegible]PLICIT [illegible]AL[illegible](D)
   CO[illegible] /CC/ DTAU2
   D1INT1=12.D0*DTAU2*(-2.D0+3.D0*DX)
   [illegible]
   [illegible]
```

```
      FUNCTION DRS0(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=1.D0-2.D0*DZ
      DRS0=4.D0*DTAU2*DS1*DS1
      RETURN
      END



      FUNCTION DSPI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=DZ*(1.D0-DZ)
      DSPI=DTAU2*DS1*DS1
      RETURN
      END



      FUNCTION DS1PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1PI=2.D0*DTAU2*DZ*(1.D0-DZ)*(1.D0-2.D0*DZ)
      RETURN
      END



      FUNCTION DS2PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
      RETURN
      END



      FUNCTION DS3PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3PI=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END



      FUNCTION DS4PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS4PI=24.D0*DTAU2
      RETURN
      END



      FUNCTION DS5PI(DZ)
      IMPLICIT REAL*8(D)
      DS5PI=0.D0
      RETURN
      END
```

```
      FUNCTION DS6PI(DZ)
      IMPLICIT REAL*8(D)
      DS6PI=0.D0
      RETURN
      END



      FUNCTION DS7PI(DZ)
      IMPLICIT REAL*8(D)
      DS7PI=0.D0
      RETURN
      END



      SUBROUTINE DHPCG(PRM,I,IDER,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1.0 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE PRESSURE DISTRIBUTIONS FOR FREE
C     STREAM MACH NUMBERS BELOW THE LOWER CRITICAL ON LIFTING PARABOLIC-
C     ARC BODIES, WHICH HAVE ELLIPTIC CROSS SECTIONS THAT MAINTAIN A
C     CONSTANT RATIO OF MAJOR TO MINOR AXES ALONG THE ENTIRE LENGTH OF
C     THE BODY, BY USING THE METHOD OF LOCAL LINEARIZATION AND THE
C     TRANSONIC EQUIVALENCE RULE--FOR REFERENCE SEE SPREITER,J.R. AND
C     ALKSNE,A.Y.,NASA TR-R2 AND HEASLET,M.A. AND SPREITER,J.R.,NASA
C     TR-1318
C
C
C*************************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON TWO CARDS AS FOLLOWS
C     FIRST CARD
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGLES OF ATTACK (IN DEGREES) WITH EACH
C     SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C     SECOND CARD
C     COLUMNS 1 TO 50--FIVE ANGULAR LOCATIONS (IN DEGREES) ON THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE AT
C     THE FIVE INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE, THETA,
C     IN THE CROSS FLOW PLANE AND FOR THE FOUR INPUTTED ANGLES OF ATTACK
C
C
C*************************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D)
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION DQ(5),DQ1(5),DQ2(5),DTH(5),DTH1(5)
      DIMENSION DALPHA(4),DALPH1(4),DSSQ(4),DSC(4),DCSQ(4),DS(4),DC(4)
      EXTERNAL FCT,OUTP
      COMMON DLL1
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /CC/ DTAU2
      COMMON /H/ DQ,DQ1,DQ2,DSSQ,DSC,DCSQ,DALPH1
      COMMON /MRMC/ DMRTE,DMRM
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      COMMON /P/ DTAU
   10 READ (5,100) DM,DL,DF,DX,DALPHA
  100 FORMAT (8(D10.2))
      READ (5,110) DTH
  110 FORMAT (5(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DM.GE.1..OR.DM.LE.0.) GO TO 306
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.1415926535897900
      DGAMMA=1.400
      DMM=DX
      DZ0=.02D0
      DZF=.98D0
      DTAU=1.0D0/DF
      DTAU2=DTAU*DTAU
      D1M=DM*DM-1.D0
      DK=(DM*DM+2.4D0
      MM=1
      DZS=.2113248654051871D0
      DLL=DLOG((DL+1.D0)+(DL+1.D0)/(4.D0*DL))
      DLL1=DLL
      DL1=DL*DL-1.D0
      DL2=DL**4-1.D0
      DL3=(DL**2.5D0)*(DL+1.D0)
      DL4=DL*(DL**3+DL+2.D0)
      DO 102 I=1,5
      DTH1(I)=DTH(I)
  102 DTH(I)=DPI*DTH(I)/180.D0
      DO 399 K=1,5
      DQ(K)=DCPTKS(DTH(K))
      DQ1(K)=DCPL1(DTH(K))
      DQ2(K)=DCPL2(DTH(K))
  399 CONTINUE
      DO 398 I=1,4
      DALPH1(I)=DALPHA(I)
      DALPHA(I)=DPI*DALPHA(I)/180.D0
      DS(I)=DSIN(DALPHA(I))
      DC(I)=DCOS(DALPHA(I))
      DSSQ(I)=DS(I)*DS(I)
      DSC(I)=DS(I)*DC(I)
  398 DCSQ(I)=DC(I)*DC(I)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DGAMMA
      WRITE (6,606) DM
      DMRM=0.
      IF(DMM-.001) 920,920,921
  921 DWRTE=DZS/DMM+1.
      MRM=DWRTE
      DWRTE=MRM
      DWRTE=DWRTE*DMM+.001
      DMRM=-DMM
  920 CONTINUE
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO NOSE//)
      WRITE (6,5) DTH1
    5 FORMAT (1H ,6X,1HX,8X,10HALPHA(DEG),5(2X,3HCP(,F6.1,4HDEG))/)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZ0
      DPRMT(3)=-.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      WRITE (6,180)
   23 WRITE (6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO TAIL//)
      WRITE (6,5) DTH1
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      GO TO 10
```

```
600 FORMAT (1H1,35X67HCALCULATION OF SURFACE PRESSURE DISTRIBUTIONS FO
   1R PURELY SUBSONIC   /36X68HFLOW ABOUT A LIFTING PARABOLIC-ARC BODY
   2, WHICH HAS AN ELLIPTIC CROSS /36X62HSECTION THAT MAINTAINS A CONS
   3TANT RATIO OF MAJOR TO MINOR AXES /36X65HALONG THE ENTIRE LENGTH O
   4F THE BODY, BY USING THE METHOD OF LOCAL/36X48HLINEARIZATION AND T
   5HE TRANSONIC EQUIVALENCE RULE///)
601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
604 FORMAT (1H ,20HS**(X) = 0 AT X/L = ,16X,D12.5)
605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
606 FORMAT (1H ,26HFREE STREAM MACH NUMBER = ,10X,D12.5)
304 WRITE (6,400)
400 FORMAT (1HO,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
    GO TO 10
305 WRITE (6,401)
401 FORMAT (1HO,54HRATIO OF MAJOR TO MINOR AXES MUST BE GREATER THAN Z
   1ERO)
    GO TO 10
306 WRITE (6,402)
402 FORMAT (113H FREE STREAM MACH NUMBER MUST BE GREATER THAN 0 AND LE
   1SS THAN THE LOWER CRITICAL MACH NUMBER WHICH IS LESS THAN 1)
    GO TO 10
307 WRITE (6,403)
403 FORMAT (1HO,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT
   1MUST BE GREATER THAN 0 AND LESS THAN 1 )
    GO TO 10
    END



    SUBROUTINE FCT(DX,DY,DZ)
    IMPLICIT REAL*8(D)
    DIMENSION DY(1),DZ(1)
    COMMON /A/ DIM,DK
    COMMON /B/ MM
    COMMON /E/ DEF
    DA=DSPI(DX)
    DA2=DS2PI(DX)
    DA3=DS3PI(DX)
    DEF=DA2*(DLOG(DA/(DX*(1.D0-DX)))+3.D0)
    DUU=-DIM-DK*(DY(1)+DEF)
    IF(DUU) 2,2,1
  1 DZ(1)=DA3*DLOG(DUU)
    RETURN
  2 MM=2
    WRITE (6,700) DX
700 FORMAT(1HO,43HLOG ARGUMENT (1-M*M-KU) IS NEGATIVE AT X = ,D12.5)
    WRITE (6,701)
701 FORMAT (1H ,72HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER GREATE
   1R THAN LOWER CRITICAL)
    RETURN
    END



    SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRMT)
    IMPLICIT REAL*8(D)
    DIMENSION DY(1),DDERY(1),DPRMT(5),DQ(5),DQ1(5),DQ2(5),DCP(5)
    DIMENSION DALPH1(4),DSSQ(4),DSC(4),DCSC(4)
    COMMON DLL1
    COMMON /E/ DEF
    COMMON /H/ DQ,DQ1,DQ2,DSSQ,DSC,DCSQ,DALPH1
    COMMON /MRMC/ DWRTE,DMRM
    IF(IHLF.GT.10) GO TO 10
    IF(DX.LE..01.OR.DX.GE..99) GO TO 30
    IF(DMRM) 50,30,60
 50 IF(DX.GT.DWRTE) RETURN
    GO TO 30
 60 IF(DX.LT.DWRTE) RETURN
 30 DU=DY(1)+DEF
    DCP1=2.D0*DU
    D2=DR(DX)
    D1=D2*D2
    D3=2.D0*DLL1*DS2PI(DX)
    WRITE (6,3)
  3 FORMAT (1HO)
    DO 1 I=1,4
    DO 2 J=1,5
    DCP(J)=-DCSQ(I)*(DCP1+D3+D1*DQ1(J))-DSC(I)*D2*DQ1(J)-DSSQ(I)*DQ2(J)
  2 CONTINUE
    WRITE (6,11) DX,DALPH1(I),DCP
 11 FORMAT (1H ,D12.5,3X,D10.3,5(3X,D12.5))
  1 CONTINUE
    DWRTE=DWRTE+DMRM
    RETURN
 10 WRITE (6,100)
100 FORMAT (1HO,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
   1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
   2ZE (.001) 10 TIMES )
    GO TO 30
    END



    FUNCTION DCPTKS(DTHETA)
    IMPLICIT REAL*8(D)
    COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
    DS=DSIN(DTHETA)
    DSSQ=DS*DS
    DCPTKS=2.D0-DL*(1.D0+DL1*DSSQ)/(1.D0+DL2*DSSQ)
    RETURN
    END



    FUNCTION DCPL1(DTHETA)
    IMPLICIT REAL*8(D)
    COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
    DS=DSIN(DTHETA)
    DSSQ=DS*DS
    D1=DSQRT(1.D0+DL1*DSSQ)
    DCPL1=2.D0*DL3*D1*DS/(1.D0+DL2*DSSQ)
    RETURN
    END



    FUNCTION DCPL2(DTHETA)
    IMPLICIT REAL*8(D)
    COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
    DS=DSIN(DTHETA)
    DSSQ=DS*DS
    DCPL2=(DL*(DL+2.D0)-DL4*DSSQ)/(1.D0+DL2*DSSQ)
    RETURN
    END



    FUNCTION DR(DZ)
    IMPLICIT REAL*8(D)
    COMMON /P/ DTAU
    DR=2.D0*DTAU*(1.D0-2.D0*DZ)
    RETURN
    END
```

```
      FUNCTION DRS0(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=1.D0-2.D0*DZ
      DRS0=4.D0*DTAU2*DS1*DS1
      RETURN
      END



      FUNCTION DSPI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=DZ*(1.D0-DZ)
      DSPI=DTAU2*DS1*DS1
      RETURN
      END



      FUNCTION DS2PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
      RETURN
      END



      FUNCTION DS3PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3PI=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END



      SUBROUTINE DHPCG(PRM1,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**A OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE PRESSURE DISTRIBUTIONS FOR FREE
C     STREAM MACH NUMBERS ABOVE THE UPPER CRITICAL ON ELLIPTIC PARABOLIC-
C     ARC BODIES, WHICH HAVE ELLIPTIC CROSS SECTIONS THAT MAINTAIN A
C     CONSTANT RATIO OF MAJOR TO MINOR AXES ALONG THE ENTIRE LENGTH OF
C     THE BODY, BY USING THE METHOD OF LOCAL LINEARIZATION AND THE
C     TRANSONIC EQUIVALENCE RULE--FOR REFERENCE SEE SPREITER,J.R. AND
C     ALKSNE,A.Y.,NASA TR-R2 AND HEASLET,M.A. AND SPREITER,J.R.,NASA
C     TR-1318
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL CONSTANTS AND ARE IN-
C     PUTTED ON TWO CARDS AS FOLLOWS
C     FIRST CARD
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGLES OF ATTACK (IN DEGREES) WITH EACH
C     SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C     SECOND CARD
C     COLUMNS 1 TO 50--FIVE ANGULAR LOCATIONS (IN DEGREES) ON THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE AT
C     THE FIVE INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE, THETA,
C     IN THE CROSS FLOW PLANE AND FOR THE FOUR INPUTTED ANGLES OF ATTACK
C
C
C*****************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D)
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(16,1)
      DIMENSION D0(5),D01(5),D02(5),DTH(5),DTH1(5)
      DIMENSION DALPHA(4),DALPH1(4),DSSO(4),DSC(4),DCSO(4),DS(4),DC(4)
      EXTERNAL FCT,OUTP
      COMMON DLL1
      COMMON /A/ D1M,DK
      COMMON /B/ MM
      COMMON /CC/ DTAU2
      COMMON /F/ DZS
      COMMON /H/ D0,D01,D02,DSSO,DSC,DCSO,DALPH1
      COMMON /MRMC/ DWRTE,DMRM
      COMMON /O/ DL,DLL,DL1,DL2,DL3,DL4
      COMMON /P/ DTAU
   10 READ (5,100) DM,DL,DF,DX,DALPHA
  100 FORMAT (8(D10.2))
      READ (5,110) DTH
  110 FORMAT (5(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DM.LE.1.) GO TO 306
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.14159265358979300
      DGAMMA=1.4D0
      DMM=DX
      DZ0=.02D0
      DZF=.98D0
      DTAU=1.D0/DF
      DTAU2=DTAU*DTAU
      D1M=DM*DM-1.D0
      DK=DM*DM*2.4D0
      MM=1
      DZS=.2113248654051871D0
      DLL=DLOG((DL+1.D0)*(DL+1.D0)/(4.D0*DL))
      DLL1=DLL
      DL1=DL*DL-1.D0
      DL2=DL**4-1.D0
      DL3=(DL**2.5D0)*(DL+1.
      DL4=DL*(DM**3+DL+2.D0)
      DO 102 I=1,5
      DTH1(I)=DTH(I)
  102 DTH(I)=DPI*DTH(I)/180.D0
      DO 399 K=1,5
      D0(K)=DCPTKS(DTH(K))
      D01(K)=DCPL1(DTH(K))
      D02(K)=DCPL2(DTH(K))
  399 CONTINUE
      DO 398 I=1,4
      DALPH1(I)=DALPHA(I)
      DALPHA(I)=DPI*DALPHA(I)/180.D0
      DS(I)=DSIN(DALPHA(I))
      DC(I)=DCOS(DALPHA(I))
      DSSO(I)=DS(I)*DS(I)
      DSC(I)=DS(I)*DC(I)
  398 DCSO(I)=DC(I)*DC(I)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DGAMMA
      WRITE (6,606) DM
      DMRM=0.
      IF(DMM-.001) 920,920,921
  921 DWRTE=DZS/DMM+1.
      MRM=DWRTE
      DWRTE=MRM
      DWRTE=DWRTE*DMM+.001
      DMRM=-DMM
  920 CONTINUE
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,101)
  101 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO NOSE//)
      WRITE (6,5) DTH1
    5 FORMAT (1H ,6X,1HX,8X,10HALPHA(DEG),5(2X,3HCP(,F6.1,4HDEG))/)
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=-.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.GT.1) GO TO 10
      WRITE (6,180)
   23 WRITE (6,103)
  103 FORMAT (1H0,44HSTART OF INTEGRATION FROM S''(X) = 0 TO TAIL//)
      WRITE (6,5) DTH1
      DY(1)=0.D0
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DPRMT(3)=.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
```

```
      GO TO 10
  600 FORMAT (1H1,35X67HCALCULATION OF SURFACE PRESSURE DISTRIBUTION FO
     1R PURELY SUPERSONIC /3AX48HFLOW ABOUT A LIFTING PARABOLIC-ARC BODY
     2, WHICH HAS AN ELLIPTIC CROSS /36X62HSECTION THAT MAINTAINS A CONS
     3TANT RATIO OF MAJOR TO MINOR AXES /36X65HALONG THE ENTIRE LENGTH
     4F THE BODY, BY USING THE METHOD OF LOCAL/36X48HLINEARIZATION AND T
     5HE TRANSONIC EQUIVALENCE RULE///)
  601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
  602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
  603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
  604 FORMAT (1H ,20HS''(X) = 0 AT X/L = ,16X,D12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
  606 FORMAT (1H ,26HFREE STREAM MACH NUMBER = ,10X,D12.5)
      WRITE (6,600)
      WRITE (6,601)
      WRITE (6,602) DR
      WRITE (6,603) DF
      WRITE (6,604) DHS
      WRITE (6,605) DG
      WRITE (6,606) DMFS
      GO TO 10
  304 WRITE (6,400)
  400 FORMAT (1H0,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
      GO TO 10
  305 WRITE (6,401)
  401 FORMAT (1H0,54HRATIO OF MAJOR TO MINOR AXES MUST BE GREATER THAN 7
     1ERO)
      GO TO 10
  306 WRITE (6,402)
  402 FORMAT (1H0,100HFREE STREAM MACH NUMBER MUST BE GREATER THAN THE U
     1PPER CRITICAL MACH NUMBER WHICH IS GREATER THAN 1 )
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (1H0,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT
     1MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END



      SUBROUTINE FCT(DX,DY,DZ)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ DIM,DK
      COMMON /B/ MM
      COMMON /E/ DHF
      COMMON /F/ DZS
      DA=DSPI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      DHF=DA2*DLOG(DA/(DX*DX))+2.D0*DINT(DX)
      DUU=D1M+DK*(DY(1)+DHF)
      IF(DUU) 2,2,1
    1 DZ(1)=DA3*DLOG(DUU)
      RETURN
    2 MM=2
      WRITE (6,700) DX
  700 FORMAT (1H0,43HLOG ARGUMENT (M**2-1+K(I)) IS NEGATIVE AT X = ,D12.5)
      IF(DX-DZS) 3,4,4
    3 WRITE (6,701)
  701 FORMAT (1H ,69HPROGRAM TERMINATED BECAUSE INPUT MACH NUMBER LESS T
     1HAN UPPER CRITICAL )
    4 RETURN
      END



      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,DPRNT)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DDERY(1),DPRNT(5),DD(5),DD1(5),DD2(5),DP(5)
      DIMENSION DALPH1(4),DSS0(4),DSC(4),DCS0(4)
      COMMON DLL1
      COMMON /E/ DHF
      COMMON /H/ DD,DD1,DD2,DSS0,DSC,DCS0,DALPH1
      COMMON /WORK/ DWRTF,DMRM
      IF(IHLF.GT.10) GO TO 10
      IF(DX.LE..01,OR.DX.GE..99) GO TO 30
      IF(DWRTF) 50,30,60
```

```
   50 IF(DX.GT.DWRTF) RETURN
      GO TO 30
   60 IF(DX.LT.DWRTF) RETURN
   30 DD=DY(1)+DHF
      DCP1=2.D0*DD
      D2=DR(DX)
      D1=D2*DZ
      D3=2.D0*DLL1*DS2PI(DX)
      WRITE (6,3)
    3 FORMAT (1HC)
      DO 1 I=1,4
      DO 2 J=1,5
      DCP(J)=-DCS0(I)*(DCP1+D3+D1*DD(J))-DSC(I)*D2*DD1(J)-DSS0(I)*DD2(J)
    2 CONTINUE
      WRITE (6,11) DX,DALPH1(I),DCP
   11 FORMAT (1H ,D12.5,3X,D10.3,5(3X,D12.5))
    1 CONTINUE
      DWRTF=DWRTF+DMRM
      RETURN
   10 WRITE (6,100)
  100 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      GO TO 30
      END



      FUNCTION DCPTKS(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      DCPTKS=2.D0-DL*(1.D0+DL1*DSS0)/(1.D0+DL2*DSS0)
      RETURN
      END



      FUNCTION DCPL1(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      D1=DSQRT(1.D0+DL1*DSS0)
      DCPL1=2.D0*DL3*D1*DS/(1.D0+DL2*DSS0)
      RETURN
      END



      FUNCTION DCPL2(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      DCPL2=(DL*(DL+2.D0)-DL4*DSS0)/(1.D0+DL2*DSS0)
      RETURN
      END



      FUNCTION DINT(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTWO
      DINT=12.D0*DTAN2*DZ*(-1.D0+1.500*DZ)
      RETURN
      END
```

```
      FUNCTION DR(DZ)
      IMPLICIT REAL*8(D)
      COMMON /P/ DTAU
      DR=2.D0*DTAU*(1.D0-2.D0*DZ)
      RETURN
      END


      FUNCTION DRSQ(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=1.D0-2.D0*DZ
      DRSQ=4.D0*DTAU2*DS1*DS1
      RETURN
      END


      FUNCTION DSPI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=DZ*(1.D0-DZ)
      DSPI=DTAU2*DS1*DS1
      RETURN
      END


      FUNCTION DS2PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
      RETURN
      END


      FUNCTION DS3PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3PI=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END


      SUBROUTINE DHPCG(PRM1,I1,DER1,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

```
C     PROGRAM FOR DETERMINING THE SURFACE PRESSURE DISTRIBUTION ON A BODY OF
C     STREAM WISE CORNERS AT AN ANGLE OF ATTACK IN SUPERSONIC FLOW
C     WHICH HAVE ELLIPTIC CROSS SECTIONS THAT VARY ALONG THE LENGTH OF THE BODY,
C     OF MAJOR TO MINOR AXES ALONG THE ENTIRE LENGTH OF THE BODY,
C     USING THE METHOD OF LOCAL LINEARIZATION AND THE TRAPEZOIDAL RULE INTEGRATION
C     RULE--FOR REFERENCE SEE SPREITER,J.R. AND ALKSNE,A.Y., NASA TR-R2
C     AND HEASLET,M.A. AND SPREITER,J.R., NASA TR-131
C
C
C*****************************************************************
C     THE INPUT DATA FOR THIS PROGRAM ARE ALL REAL DATA AND ARE TO BE
C     PUTTED ON TWO CARDS AS FOLLOWS
C     FIRST CARD
C     COLUMNS 1 TO 10--FREE STREAM MACH NUMBER
C     COLUMNS 11 TO 20--RATIO OF MAJOR TO MINOR AXES OF ELLIPTIC CROSS
C     SECTION
C     COLUMNS 21 TO 30--FINENESS RATIO OF EQUIVALENT BODY OF REVOLUTION
C     COLUMNS 31 TO 40--INTERVAL SIZE AS FRACTION OF BODY LENGTH FOR
C     PRESSURE DISTRIBUTION PRINT-OUT
C     COLUMNS 41 TO 80--FOUR ANGLES OF ATTACK (IN DEGREES) WITH EACH
C     SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C     SECOND CARD
C     COLUMNS 1 TO 50--FIVE ANGULAR LOCATIONS (IN DEGREES) ON THE BODY
C     SURFACE WITH EACH SUCCESSIVE ANGLE OCCUPYING A SPACE OF 10 COLUMNS
C
C
C     THE OUTPUT PRESSURE DISTRIBUTIONS ARE GIVEN ON THE BODY SURFACE AT
C     THE FIVE INPUTTED ANGULAR LOCATIONS OF THE AZIMUTHAL ANGLE, THETA,
C     IN THE CROSS FLOW PLANE AND FOR THE FOUR INPUTTED ANGLES OF ATTACK
C
C
C*****************************************************************
C
C
C     MAIN PROGRAM
      IMPLICIT REAL*8(D)
      DIMENSION DPRMT(5),DY(1),DDERY(1),DAUX(1A,1)
      DIMENSION DD(5),DD1(5),DD2(5),DTH(5),DTH1(5)
      DIMENSION DALPHA(4),DALPH1(4),DSSO(4),DSC(4),DCSO(4),DS(4),DC(4)
      EXTERNAL FCT,OUTP
      COMMON DLL1
      COMMON /A/ DM,DK,D1,D1M,D2M,DZS,DU1,DU2,DU3,DU4,DU5
      COMMON /B/ MM
      COMMON /BB/ N
      COMMON /CC/ DTAU2
      COMMON /E/ DX11,DX12,DUU1,DUU2
      COMMON /F/ DV,DV3
      COMMON /H/ DD,DD1,DD2,DSSO,DSC,DCSO,DALPH1
      COMMON /PRNC/ DMRTE,DMRM
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      COMMON /P/ DTAU
   10 READ (5,100) DR,DL,DF,DX,DALPHA
  100 FORMAT (8(D10.2))
      READ (5,110) DTF
  110 FORMAT (5(D10.2))
      WRITE (6,600)
      IF(DF.LE.0.) GO TO 304
      IF(DL.LE.0.) GO TO 305
      IF(DX.LE.0..OR.DX.GE.1.) GO TO 307
      DPI=3.1415926535897932D00
      DGAM=1.4D0
      DM=DR
      DTAU=1.0D0/DF
      DTAU2=DTAU*DTAU
      DK=DM*DM-2.D00
      D1=DK+1.7D07241799019D00
      D1M=DM*DM-1.D0
      D2M=D1M/DK
      DZS=.211324865405187D00
      DZ0=.0200
      DZF=.98D0
      DX12=0.D0
      DUU2=0.D0
      N=1
      DLL=DLOG((DL+1.D0)+(DL+1.D0)/(4.D0*DL))
      DLL1=DLL
      DL1=DL+1.D0
      DL2=DL+4-1.D0
      DL3=(DL+2.D00)*(DL+1.D0)
      DL4=DL*(DL+3+DL+2.D0)
      DO 102 I=1,5
      DTH1(I)=DTH(I)
  102 DTH(I)=DPI*DTH(I)/180.D0
      DO 399 K=1,5
      DD(K)=DCOS(DTH(K))
      DD1(K)=DSIN(DTH(K))
      DD2(K)=DCOS(DTH(K))
  399 CONTINUE
      DO 396 I=1,4
      DALPH1(I)=DALPHA(I)
      DALPHA(I)=DPI*DALPHA(I)/180.D0
      DS(I)=DSIN(DALPHA(I))
      DC(I)=DCOS(DALPHA(I))
      DSSO(I)=DS(I)*DS(I)
      DSC(I)=DS(I)*DC(I)
  396 DCSO(I)=DC(I)*DC(I)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,601)
      WRITE (6,602) DL
      WRITE (6,603) DF
      WRITE (6,604) DZS
      WRITE (6,605) DCALPHA
      WRITE (6,606) DR
      DMRM =0.
      IF(D1M-.001) 920,920,921
  921 DMRTE=DZS/D1M+1.
      DMRM=DMRTE
      DMRTE=DMRTE*DMRM+.001
      DMRM=-DMRM
  920 CONTINUE
C
C     CALCULATION OF AREA AND DERIVATIVES OF AREA AT XS
C
      DA=DSPI(DZS)
      DA1=DS1PI(DZS)
      DA2=DS2PI(DZS)
      DA3=DS3PI(DZS)
      DA4=DS4PI(DZS)
      DA5=DS5PI(DZS)
      DA6=DS6PI(DZS)
      DA7=DS7PI(DZS)
      DAM=DLOG(D1M*DA/DZS)
      DU1=DA1/DA
      DU3=DA3/DA
      DU4=DA4/DA
      DU42=DU4*DU4
      DU43=DU42*DU4
C
C     CALCULATION OF INITIAL VALUE OF U
C
      DUU=DU1+DU1T(DZS)
C
C     CALCULATION OF THE 1ST DERIVATIVE OF U
C
      DU1=12.D0*DTAU2*(-1.D0+2.D0*DZS)
      DUU1=5.D-1
      N=1
   13 DU11=DU1-DU1*(DU1-DA3*(1.D0*(DU1)+DU**)-DU1)/(DU1-DA3)
```

```
      IF(DABS(DU11-DU1).LT.1.D-10) GO TO 14
      DU1=DU11
      N=N+1
      IF(N.GT.11) GO TO 200
      GO TO 13
  200 WRITE (6,104)
  104 FORMAT (1HO,4X73HPROGRAM TERMINATED BECAUSE INITIAL DERIVATIVE OF
     1U AT S''(X) = 0 REQUIRED /5X68HFOR TAYLOR SERIES START AT THAT POI
     2NT CANNOT BE DETERMINED TO WITHIN /5X36HSUFFICIENT ACCURACY IN 10
     3ITERATIONS)
      IF(N.GT.11) GO TO 10
   14 DU1=DU11
C
C     CALCULATION OF THE INITIAL 2ND DERIVATIVE OF U
C
      D2I=36.D0*DTAU2
      DA31=DA3/DU1
      DAM1=DAM+DLOG(DU1)
      DU2=(DA4*DAM1+2.D0*DA3*(D1N*(1.D0-DA31)-1.D0/DZS)+D2I)/(1.D0-2.D0
     1*DA31)
C
C     CALCULATION OF THE INITIAL 3RD DERIVATIVE OF U
C
      D3I=0.D0
      D2I=DU2/DU1-DA31*D1N
      D2C=D2I+D1N-1.D0/DZS
      DA41=DA4/DU1
      D311=D1N*(DA41-2.D0*DA31*D1N)
      D3C1=-D2I*D2I-D1N2+1.D0/(DZS*DZS)
      DU3=(DA5*DAM1+3.D0*DA4*D2C+3.D0*DA3*(D3C1-D311)+D3I)/(1.D0-3.D0*
     1DA31)
C
C     CALCULATION OF THE INITIAL 4TH DERIVATIVE OF U
C
      D4I=0.D0
      D31=DU3/DU1-D311
      D3C=D31+D3C1
      D411=(D1N*(DA5-3.D0*DA4*D1N)+D3N*(6.D0*DA1*D1N2+DA3))/DU1
      D4C1=-3.D0*D31*D31*D21+2.D0*(D21**3)+D3N+2.D0*D1N2-2.D0/(DZS**3)
      DU4=(DA6*DAM1+4.D0*DA5*D2C+6.D0*DA4*D3C+4.D0*DA3*(-D411+D4C1)+D4I
     1)/(1.D0-4.D0*DA31)
C
C     CALCULATION OF THE INITIAL 5TH DERIVATIVE OF U
C
      D5I=0.D0
      D511=(D1N*(DA6-4.D0*DA5*D1N)+4.D0*DA4*D1N3+(D3N+2.D0*D1N3)*(4.D0
     1*DA4-12.D0*DA3*D1N))/DU1
      D41=DU4/DU1-D411
      D4C=D41+D4C1
      DU5=(DA7*DAM1+5.D0*DA6*D2C+10.D0*DA5*D3C+10.D0*DA4*D4C+5.D0*DA3*
     1(-D511-4.D0*D41*D21+12.D0*D31*D21*D21-3.D0*D31*D31-6.D0*(D21**4)+
     2D4N-4.D0*D3N*D1N-6.D0*D1N2+6.D0/(DZS**4))+D5I)/(1.D0-5.D0*DA31)
C
C     START OF INTEGRATION PROCEDURE
C
      WRITE (6,180)
  180 FORMAT (1H )
      WRITE (6,99)
   99 FORMAT (1HO,44HSTART OF INTEGRATION FROM S''(X) = 0 TO NOSE//)
      WRITE (6,5) DTH1
    5 FORMAT (1H ,6X,1HX,8X,10HALPHA(DEG),5(2X,3HCP(,F6.),4HDEG))/)
      DY(1)=DU
      DDERY(1)=1.D0
      DPRMT(1)=DZS
      DPRMT(2)=DZO
      DPRMT(3)=-.001D0
      DPRMT(4)=1.D-6
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      WRITE (6,180)
   25 WRITE (6,99)
      WRITE (6,5) DTH1
      DPRMT(1)=DZS
      DPRMT(2)=DZF
      DY(1)=DU
      IF(DMRM.EQ.0.) GO TO 922
      MRM=MRM-1
      DWRTE=MRM
      DWRTE=DWRTE*DMM
      DMRM=DMM
  922 CONTINUE
   27 DPRMT(3)=.001D0
      DPRMT(4)=1.D-6
      DDERY(1)=1.D0
      NDIM=1
      CALL DHPCG(DPRMT,DY,DDERY,NDIM,IHLF,FCT,OUTP,DAUX)
      IF(IHLF.GT.10) GO TO 10
      IF(MM.EQ.2) GO TO 29
      IF(MM.EQ.3) GO TO 30
   28 GO TO 10
   29 DY(1)=DV
      DPRMT(1)=DV3
      MM=1
      WRITE (6,5) DTH1
      GO TO 27
   30 DSL=(DX12-DX11)/(DUU1-DUU2)
      DSH=DSL*(DUU2+D2M)
      DX3=DX12+DSH
      DX4=2.D0*DX3-DX12
      WRITE (6,205)
  205 FORMAT (1HO,29HSTART OF SUBSONIC CALCULATION)
      WRITE (6,509) DX3
  509 FORMAT (1HO,35HLOG ARGUMENT (M*M-1+KU) = 0 AT X = ,D12.5)
      WRITE (6,510) DX4
  510 FORMAT (1HO,35HSUBSONIC CALCULATION BEGINS AT X = ,D12.5//)
      DY(1)=-DUU2-2.D0*D2M
      DPRMT(1)=DX4
      MM=1
      WRITE (6,5) DTH1
      GO TO 27
  600 FORMAT (1H1,35X64HCALCULATION OF SURFACE PRESSURE DISTRIBUTIONS FO
     1R FLOW WITH FREE  /36X61HSTREAM MACH NUMBER AT OR NEAR 1 ABOUT A L
     2IFTING PARABOLIC-ARC /36X58HBODY, WHICH HAS AN ELLIPTIC CROSS SECT
     3ION THAT MAINTAINS A /36X61HCONSTANT RATIO OF MAJOR TO MINOR AXES
     4ALONG THE ENTIRE LENGTH /36X59HOF THE BODY, BY USING THE METHOD OF
     5 LOCAL LINEARIZATION AND /36X30HTHE TRANSONIC EQUIVALENCE RULE///)
  601 FORMAT (1H ,35HBODY AND FLOW FIELD CHARACTERISTICS//)
  602 FORMAT (1H ,31HRATIO OF MAJOR TO MINOR AXES = ,5X,D12.5)
  603 FORMAT (1H ,36HFINENESS RATIO OF EQUIVALENT BODY = ,D12.5)
  604 FORMAT (1H ,20HS''(X) = 0 AT X/L = ,16X,D12.5)
  605 FORMAT (1H ,26HRATIO OF SPECIFIC HEATS = ,10X,D12.5)
  606 FORMAT (1H ,26HFREE STREAM MACH NUMBER = ,10X,D12.5)
  304 WRITE (6,400)
  400 FORMAT (1HO,40HFINENESS RATIO MUST BE GREATER THAN ZERO)
      GO TO 10
  305 WRITE (6,401)
  401 FORMAT (1HO,54HRATIO OF MAJOR TO MINOR AXES MUST BE GREATER THAN Z
     1ERO)
      GO TO 10
  307 WRITE (6,403)
  403 FORMAT (1HO,89H INTERVAL SIZE FOR PRESSURE DISTRIBUTION PRINT-OUT
     1MUST BE GREATER THAN 0 AND LESS THAN 1 )
      GO TO 10
      END
```

```
      SUBROUTINE FCT (DX,DY,DZ)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DZ(1)
      COMMON /A/ D4,DK,D1,D1M,D2M,DZS,DU1,DU2,DU3,DU4,DU5
      COMMON /B/ MM
      COMMON /BB/ M
      COMMON /E/ DV,DV3
      IF(DABS(DX-DZS).LE.5.D-3) GO TO 10
      DA=DSPI(DX)
      DA1=DS1PI(DX)
      DA2=DS2PI(DX)
      DA3=DS3PI(DX)
      IF(M.EQ.1) GO TO 1
      IF(M.EQ.2) GO TO 2
      IF(M.EQ.3) GO TO 3
    1 IF(DX-DZS) 20,20,30
   20 DZ(1)=DA1*DA2/DA+DEXP((DY(1)+D2M-DINT(DX)-DA2*DLOG(D1*DA/DX))/DA2)
      RETURN
   30 DUU=D1M+DK*DY(1)
      IF(DUU) 20,20,40
   40 DZH=DA3*DLOG(DUU*DA/(DX*DX))+DA2*(DA1/DA-2.D0/DX)+2.D0*D1INT(DX)
      DZ(1)=DA1*DA2/DA+DEXP((DY(1)+D2M-DINT(DX)-DA2*DLOG(D1*DA/DX))/DA2)
      DIFF=DZH-DZ(1)
      IF(DIFF) 60,60,50
   50 RETURN
   60 M=2
      MM=2
      WRITE (6,199)
  199 FORMAT (1H0,31HSTART OF SUPERSONIC CALCULATION)
      WRITE (6,200) DX
  200 FORMAT (1H0,37HSUPERSONIC CALCULATION STARTS AT X = ,D12.5//)
      DV=DY(1)
      DV3=DX
      RETURN
    2 DUU=D1M+DK*DY(1)
      IF(DUU) 80,80,70
   70 DZ(1)=DA3*DLOG(DUU*DA/(DX*DX))+DA2*(DA1/DA-2.D0/DX)+2.D0*D1INT(DX)
      RETURN
   80 M=3
      MM=3
      RETURN
    3 DUU=-D1M-DK*DY(1)
      DZ(1)=DA3*DLOG(DUU*DA/(DX*(1.D0-DX)))+DA2*(DA1/DA-1.D0/DX+1.D0
     1/(1.D0-DX))+D1INT(DX)+D1INT1(DX)
      RETURN
   10 DZ(1)=DU1+DU2*(DX-DZS)+DU3*((DX-DZS)**2)/2.D0+DU4*((DX-DZS)**3)/
     16.D0+DU5*((DX-DZS)**4)/24.D0
      RETURN
      END
```

```
      SUBROUTINE OUTP(DX,DY,DDERY,IHLF,NDIM,PRMT)
      IMPLICIT REAL*8(D)
      DIMENSION DY(1),DDERY(1),PRMT(5),D0(5),D01(5),D02(5),DCP(5)
      DIMENSION DALPH1(4),DSS0(4),DSC(4),DCS0(4)
      COMMON DLL1
      COMMON /BB/ M
      COMMON /E/ DX11,DX12,DUU1,DUU2
      COMMON /H/ D0,D01,D02,DSS0,DSC,DCS0,DALPH1
      COMMON /WRMC/ DWRTE,DMRM
      DX11=DX12
      DX12=DX
      DUU1=DUU2
      DUU2=DY(1)
      IF(IHLF.GT.10) GO TO 10
      IF(DX.LE..01.OR.DX.GE..99) GO TO 30
      IF(DMRM) 50,30,60
   50 IF(DX.GT.DWRTE) RETURN
      GO TO 30
   60 IF(DX.LT.DWRTE) RETURN
   30 DCP1=2.D0*DY(1)
      D2=DR(DX)
      D1=D2*D2
      D3=2.D0*DLL1*DS2PI(DX)
      WRITE (6,4)
    4 FORMAT (1H0)
      DO 1 I=1,4
      DO 2 J=1,5
      DCP(J)=-DCS0(I)*(DCP1+D3+D1*D0(J))-DSC(I)*D2*D01(J)-DSS0(I)*DQ2(J)
    2 CONTINUE
      WRITE (6,11) DX,DALPH1(I),DCP
   11 FORMAT (1H ,D12.5,3X,D10.3,5(3X,D12.5))
    1 CONTINUE
      DWRTE=DWRTE+DMRM
      RETURN
   10 WRITE (6,100)
  100 FORMAT (1H0,74HINTEGRATION TERMINATED BECAUSE ACCUMULATED ERRORS H
     1AVE CAUSED INTEGRATION /1X55HSUBROUTINE TO BISECT ORIGINAL STEP SI
     2ZE (.001) 10 TIMES )
      GO TO 30
      END
```

```
      FUNCTION DCPTKS(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      DCPTKS=2.D0-DL*(1.D0+DL1*DSS0)/(1.D0+DL2*DSS0)
      RETURN
      END
```

```
      FUNCTION DCPL1(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      D1=DSQRT(1.D0+DL1*DSS0)
      DCPL1=2.D0*DL3*D1*DS/(1.D0+DL2*DSS0)
      RETURN
      END
```

```
      FUNCTION DCPL2(DTHETA)
      IMPLICIT REAL*8(D)
      COMMON /D/ DL,DLL,DL1,DL2,DL3,DL4
      DS=DSIN(DTHETA)
      DSS0=DS*DS
      DCPL2=(DL*(DL+2.D0)-DL4*DSS0)/(1.D0+DL2*DSS0)
      RETURN
      END
```

```
      FUNCTION DINT(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DINT=12.D0*DTAU2*DZ*(-1.D0+1.500*DZ)
      RETURN
      END
```

```
      FUNCTION D1INT1(DX)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      D1INT1=12.D0*DTAU2*(.500-2.D0*DX+1.500*D0*DX)*DX)
      RETURN
      END


      FUNCTION D1INT(DX)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      D1INT=12.D0*DTAU2*(-1.D0+3.D0*DX)
      RETURN
      END


      FUNCTION D1INT1(DX)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      D1INT1=12.D0*DTAU2*(-2.D0+3.D0*DX)
      RETURN
      END


      FUNCTION DR(DZ)
      IMPLICIT REAL*8(D)
      COMMON /P/ DTAU
      DR=2.D0*DTAU*(1.D0-2.D0*DZ)
      RETURN
      END


      FUNCTION DRS0(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=1.D0-2.D0*DZ
      DRS0=4.D0*DTAU2*DS1*DS1
      RETURN
      END


      FUNCTION DSPI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1=DZ*(1.D0-DZ)
      DSPI=DTAU2*DS1*DS1
      RETURN
      END


      FUNCTION DS1PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS1PI=2.D0*DTAU2*DZ*(1.D0-DZ)*(1.D0-2.D0*DZ)
      RETURN
      END
```

```
      FUNCTION DS2PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS2PI=2.D0*DTAU2*(1.D0-6.D0*DZ*(1.D0-DZ))
      RETURN
      END


      FUNCTION DS3PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS3PI=12.D0*DTAU2*(2.D0*DZ-1.D0)
      RETURN
      END


      FUNCTION DS4PI(DZ)
      IMPLICIT REAL*8(D)
      COMMON /CC/ DTAU2
      DS4PI=24.D0*DTAU2
      RETURN
      END


      FUNCTION DS5PI(DZ)
      IMPLICIT REAL*8(D)
      DS5PI=0.D0
      RETURN
      END


      FUNCTION DS6PI(DZ)
      IMPLICIT REAL*8(D)
      DS6PI=0.D0
      RETURN
      END


      FUNCTION DS7PI(DZ)
      IMPLICIT REAL*8(D)
      DS7PI=0.D0
      RETURN
      END


      SUBROUTINE DHPCG(PRMT,Y,DERY,NDIM,IHLF,FCT,OUTP,AUX)
C     THE INTEGRATING SUBROUTINE DHPCG (DOUBLE PRECISION VERSION) USED WITH
C     THIS PROGRAM IS THE SAME AS USED IN THE PROGRAM FOR CALCULATING
C     SURFACE AND FLOW FIELD PRESSURE DISTRIBUTIONS FOR FREE STREAM MACH
C     NUMBERS AT OR NEAR 1 ON NONLIFTING BODIES OF REVOLUTION HAVING OR-
C     DINATES R PROPORTIONAL TO X-X**N OR 1-X-(1-X)**N
C     FOR A LISTING OF THIS SUBROUTINE SEE THAT PROGRAM
```

REFERENCES

1. Spreiter, J. R., Stahara, S. S., and Frey, W. H.: Calculative Techniques for Transonic Flows. NASA SP-228, 1969, pp. 53-73.

2. Spreiter, J. R. and Stahara, S. S.: Developments in Transonic Flow Theory. Zeits. für Flugwiss., Bd. 18, Heft 2/3, 1970, pp. 33-40.

3. Truitt, R. W.: Shockless Transonic Airfoils. AIAA Paper No. 70-187, presented at AIAA 8th Aerospace Sciences Meeting, New York, N. Y., Jan. 19-21, 1970.

4. Magnus, R. and Yoshihara, H.: Inviscid Transonic Flow Over Airfoils. AIAA Paper No. 70-47, presented at AIAA 8th Aerospace Sciences Meeting, New York, N. Y., Jan. 19-21, 1970.

5. Murman, E. M. and Cole, J. D.: Calculation of Plane Steady Transonic Flows. AIAA Paper No. 70-188, presented at AIAA 8th Aerospace Sciences Meeting, New York, N. Y., Jan. 19-21, 1970.

6. Heaslet, M. A. and Spreiter, J. R.: Three-Dimensional Transonic Flow Theory Applied to Slender Wings and Bodies. NACA Rep. 1318, 1957.

7. Spreiter, J. R. and Alksne, A.: Slender Body Theory Based on Approximate Solution of the Transonic Flow Equation. NASA TR R-2, 1959.

8. McDevitt, J. B. and Taylor, R. A.: Force and Pressure Measurements at Transonic Speeds for Several Bodies Having Elliptical Cross Sections. NACA TN 4362, 1958.

9. Spreiter, J. R.: Aerodynamics of Wings and Bodies at Transonic Speeds. J. Aero/Space Sci., vol. 26, no. 8, Aug. 1969, pp. 465-487.

10. Spreiter, J. R., Smith, D. W., and Hyett, B. J.: A Study of the Simulation of Flow with Free-Stream Mach Number 1 in a Choked Wind Tunnel. NASA TR R-73, 1960.

11. Oswatitsch, K.: Die Geschwindigkeitsverteilung bei lokalen Überschallgebieten an flachen Profilen. ZAMM, Bd. 30, Heft 1/2, Jan./Feb. 1950, pp. 17-24.

12. Oswatitsch, K.: Die Geschwindigkeitsverteilung an symmetrischen Profilen beim Auftreten lokaler Überschallgebiete. Acta Physica Austriaca, Bd. 4, Heft 2/3, Dec. 1950, pp. 228-271.

13. Gullstrand, T. R.: The Flow Over Symmetrical Aerofoils without Incidence in the Lower Transonic Range. KTH Aero TN 20, Royal Inst Tech., Stockholm, Sweden, 1951.

14. Gullstrand, T. R.: The Flow Over Symmetrical Aerofoils without Incidence at Sonic Speed. KTH Aero TN 24, Royal Inst. Tech., Stockholm, Sweden, 1952.

15. Gullstrand, T. R.: A Theoretical Discussion of Some Properties of Transonic Flow Over Two-Dimensional Symmetrical Aerofoils at Zero Lift with a Simple Method to Estimate the Flow Properties. KTH Aero TN 25, Royal Inst. Tech., Stockholm, Sweden, 1952.

16. Gullstrand, T. R.: The Flow Over Two-Dimensional Aerofoils at Incidence in the Transonic Speed Range. KTH Aero TN 27, Royal Inst. Tech., Stockholm, Sweden, 1952.

17. Spreiter, J. R. and Alksne, A.: Theoretical Prediction of Pressure Distributions on Nonlifting Airfoils at High Subsonic Speeds. NACA Rep. 1217, 1955. (Supersedes NACA TN 3096.)

18. Spreiter, J. R., Alksne, A., and Hyett, B. J.: Theoretical Pressure Distributions for Several Related Nonlifting Airfoils at High Subsonic Speeds. NACA TN 4148, 1958.

19. Spreiter, J. R. and Alksne, A.: Thin Airfoil Theory Based on Approximate Solution of the Transonic Flow Equation. NACA Rep. 1359, 1958. (Supersedes NACA TN 3970.)

20. Alksne, A. Y. and Spreiter, J. R.: Theoretical Pressure Distributions on Wings of Finite Span at Zero Incidence for Mach Numbers Near 1. NASA TR R-88, 1961.

21. Spreiter, J. R.: Theoretical and Experimental Analysis of Transonic Flow Fields. NACA-University Conference on Aerodynamics, Construction, and Propulsion, Vol. II, "Aerodynamics," 1954, pp. 18/1-18/17.

22. Guderley, K. G.: Theory of Transonic Flow. Pergamon Press, Oxford, England, 1962.

23. Ferrari, C. and Tricomi, F.: Transonic Aerodynamics. Academic Press, N. Y., 1968.

24. Oswatitsch, K., ed.: Symposium Transsonicum. Springer-Verlag, Berlin/Göttingen/Heidelberg, 1964.

25. Oswatitsch, K. and Keune, F.: The Flow Around Bodies of Revolution at Mach Number 1. *Proc. Conf. on High-Speed* Aeronautics, Polytechnic Institute of Brooklyn, Brooklyn, N. Y., Jan. 20-22, 1955, pp. 113-131.

26. Michel, R., Marchaud, F., and Le Gallo, J.: Étude des écoulements transsoniques autour des profils lenticulaires, a incidence nulle. O.N.E.R.A. Pub. No. 65, 1953.

27. Spreiter, J. R.: On the Application of Transonic Similarity Rules to Wings of Finite Span. NACA Rep. 1153, 1953. (Supersedes NACA TN 2726.)

28. Liepmann, H. W. and Roshko, A.: Elements of Gasdynamics. John Wiley and Sons, Inc., New York, N. Y., 1957, pp. 202-205.

29. Oswatitsch, K.: Die Theoretischen Arbeiten Über Schallnahe Strömungen am Flugtechnischen Institut der Kungl. Tekniska Hökskolan, Stockholm. Proc. Eighth Int. Cong. on Theo. and Appl. Mech., 1953.

30. Whitcomb, R. T.: A Study of the Zero-Lift Drag-Rise Characteristics of Wing-Body Combinations Near the Speed of Sound. NACA Rep. 1273, 1956. (Supersedes NACA TM L52H08.)

31. Spreiter, J. R.: The Aerodynamic Forces on Slender Plane and Cruciform-Wing and Body Combinations. NACA Rep. 962, 1950.

32. Heaslet, M. A., Lomax, H., and Spreiter, J. R.: Linearized Compressible-Flow Theory for Sonic Flight Speeds. NACA Rep. 956, 1950.

33. Nielsen, J. N.: Missile Aerodynamics. McGraw-Hill Book Co., Inc., New York, N. Y., 1960, p. 30.

34. Karamcheti, K.: Principles of Ideal-Fluid Aerodynamics. John Wiley and Sons, Inc., New York/London/Sydney, 1966, pp. 583-584.

35. Taylor, R. A. and McDevitt, J. B.: Pressure Distributions at Transonic Speeds for Parabolic-Arc Bodies of Revolution Having Fineness Ratios of 10, 12, and 14. NACA TN 4234, 1958.

36. McDevitt, J. B. and Taylor, R. A.: Pressure Distributions at Transonic Speeds for Slender Bodies Having Various Axial Locations of Maximum Diameter. NACA TN 4280, 1958.

37. Guderley, K. G.: Theorie schallnaher Strömungen. Springer-Verlag, Berlin/Göttingen/Heidelberg, 1957.

38. Oswatitsch, K. and Berndt, S. B.: Aerodynamic Similarity at Axisymmetric Transonic Flow Around Slender Bodies. KTH Aero TN 15, 1950, Royal Inst. Tech., Stockholm, Sweden.

39. Oswatitsch, K.: Die Berechnung wirbelfreier achsensymmetrischer Überschallfelder. Österreichisches Ingenieur-Archiv, Band X, Heft 4, 1956, pp. 359-382.

40. Schlichting, H.: Boundary Layer Theory. McGraw-Hill Book Co., Inc., New York, N. Y., 1962, pp. 18-21.

41. Page, W. A.: Experimental Study of the Equivalence of Transonic Flow About Slender Cone-Cylinders of Circular and Elliptic Cross Section. NACA TN 4233, 1958.

42. Spreiter, J. R. and Sacks, A. H.: A Theoretical Study of the Aerodynamics of Slender Cruciform-Wing Arrangements and Their Wakes. NACA Rep. 1296, 1957.

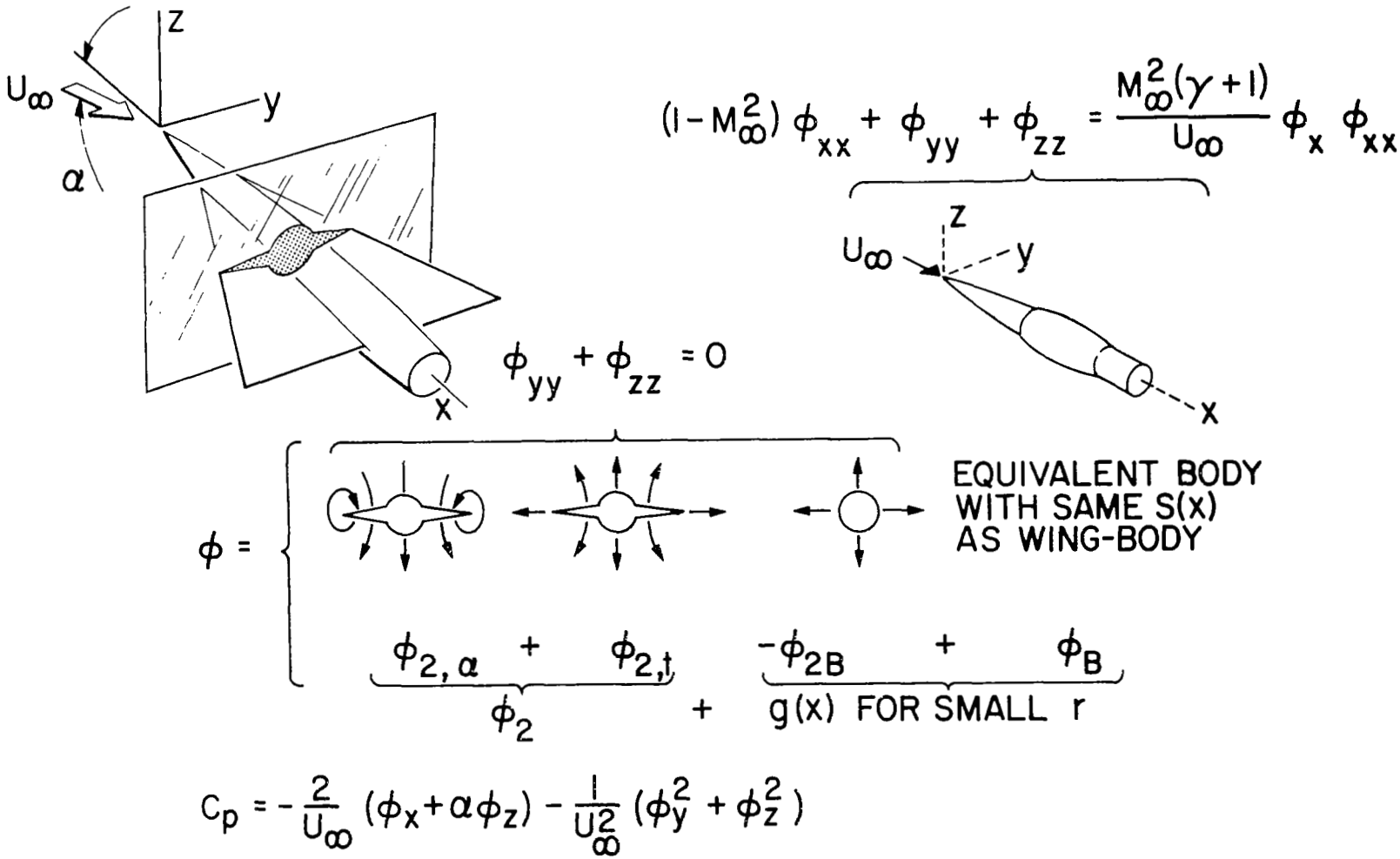


Figure 1.- Transonic equivalence rule for slender wing-body combinations.

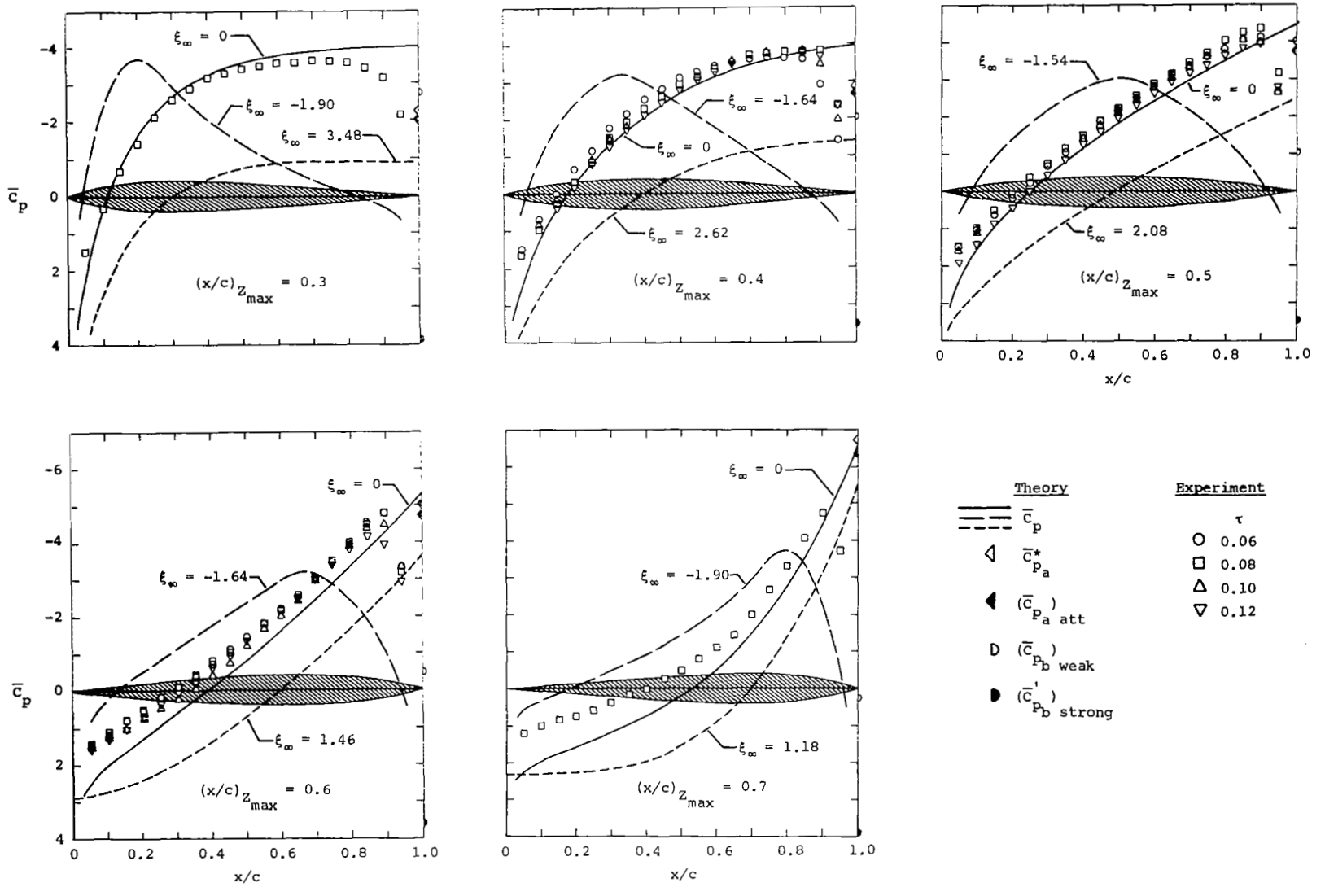


Figure 2.- Theoretical pressure distributions at $M_\infty = 1$ and at the lower and upper critical Mach numbers on a series of airfoils having various axial locations of maximum thickness; also shown are experimental data for $M_\infty = 1$.

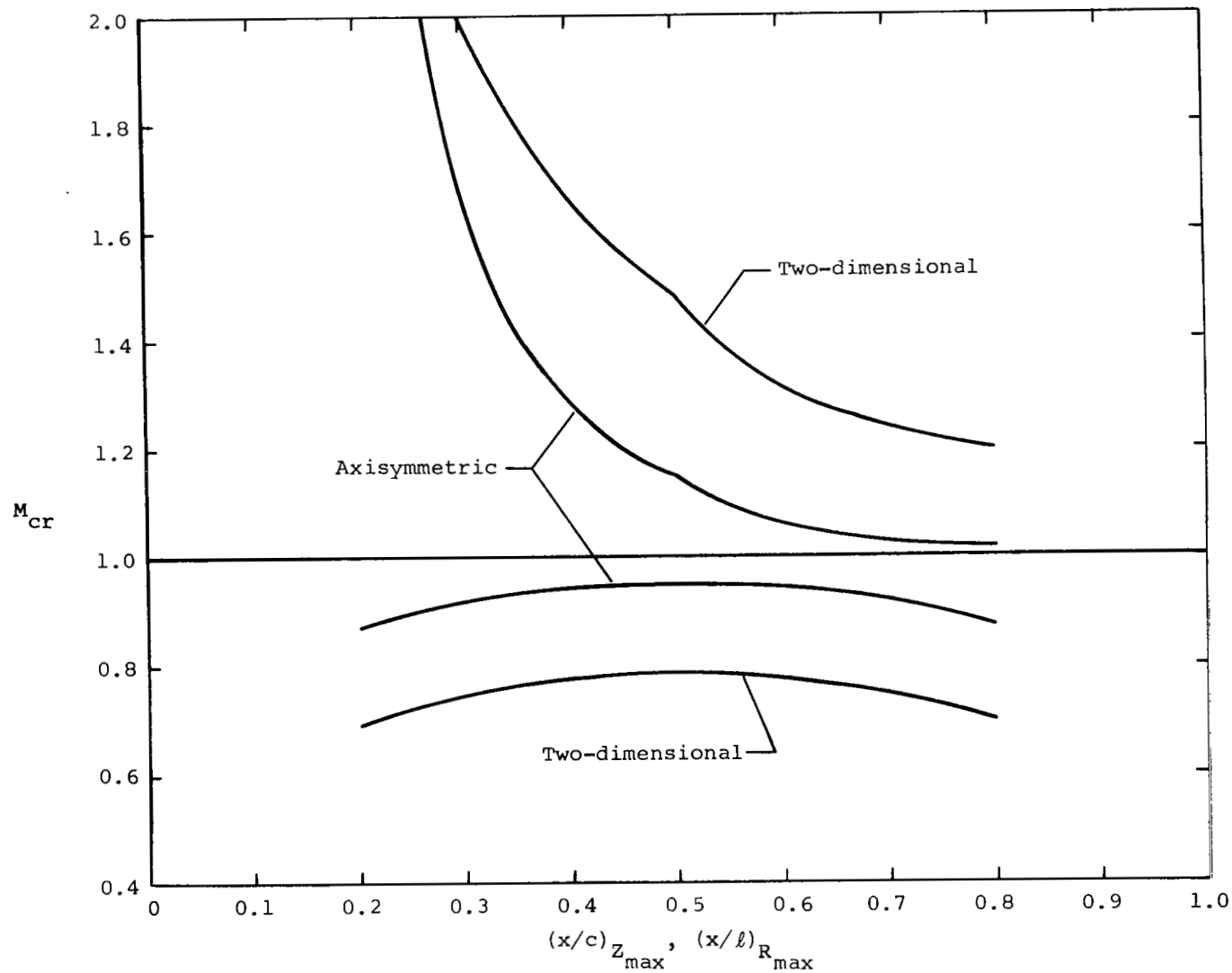
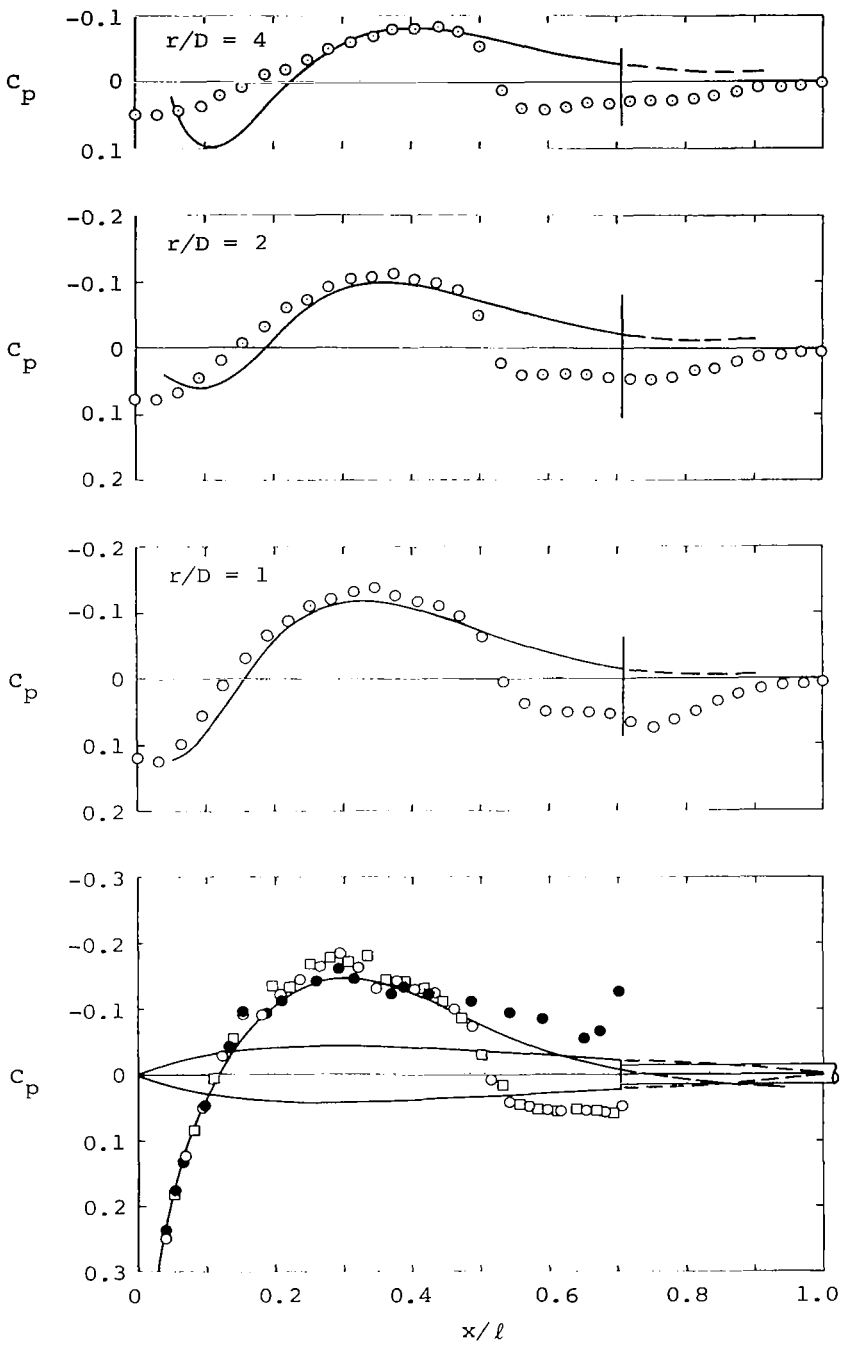
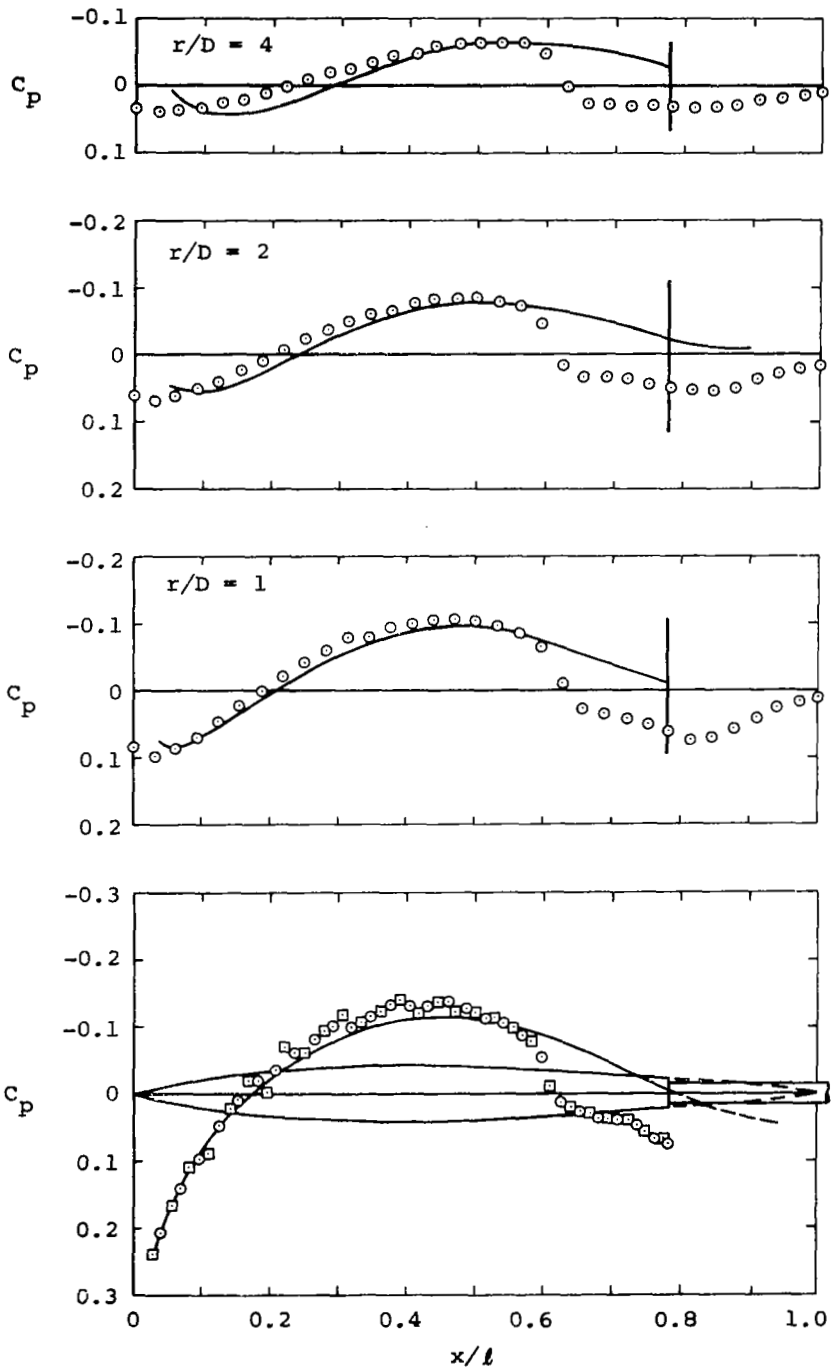


Figure 3.- Upper and lower critical Mach numbers versus location of maximum thickness for two-dimensional and axisymmetric bodies having thickness ratio τ and D/ℓ, respectively, of 1/12 and profiles described by equations (6), (8), (76), and (78).



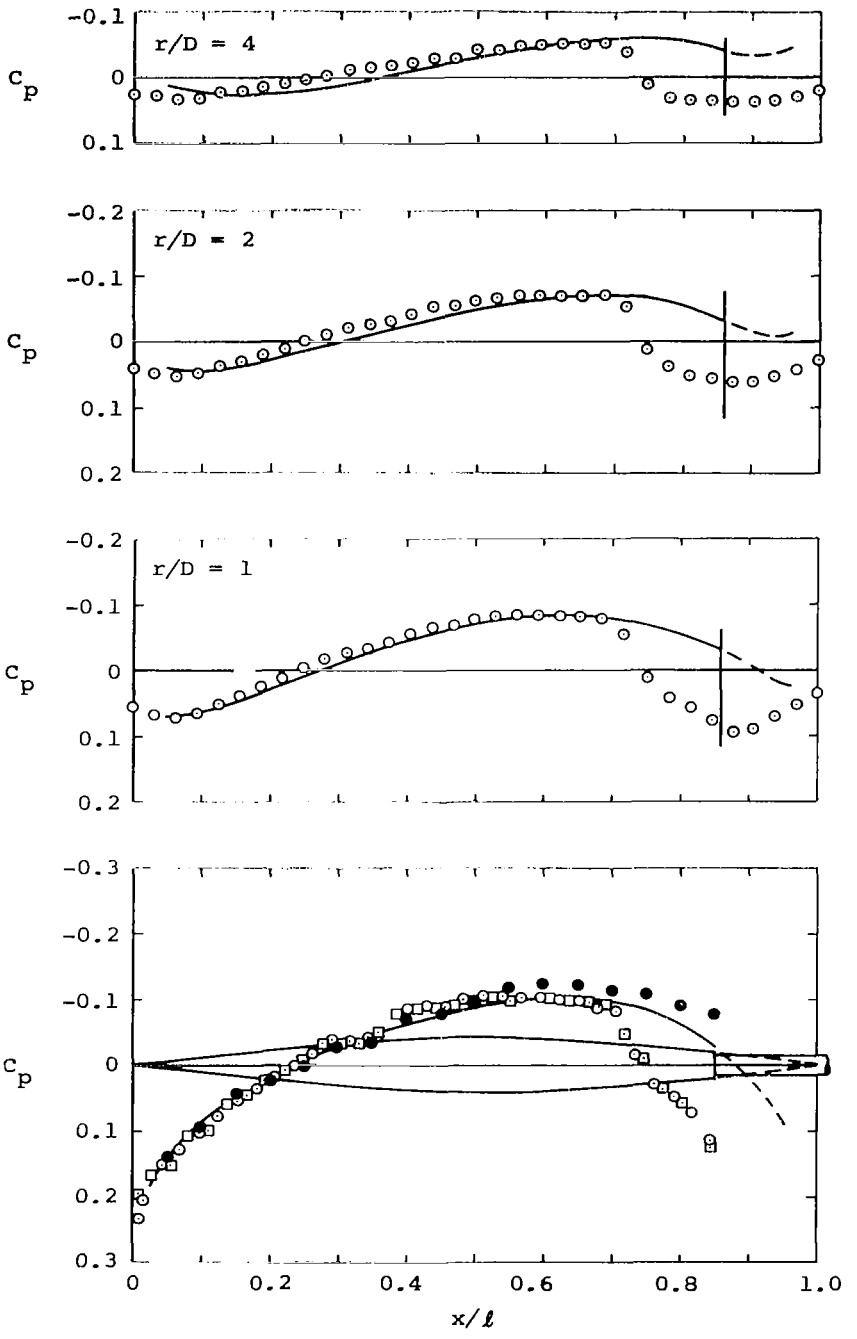
(a) $(x/\ell)_{R_{max}} = 0.3$.

Figure 4.- Theoretical and experimental surface and flow field pressure distributions at $M_\infty = 1$ for several members of a family of bodies of revolution having thickness ratio $D/\ell = 1/12$ and various locations of the point of maximum thickness.



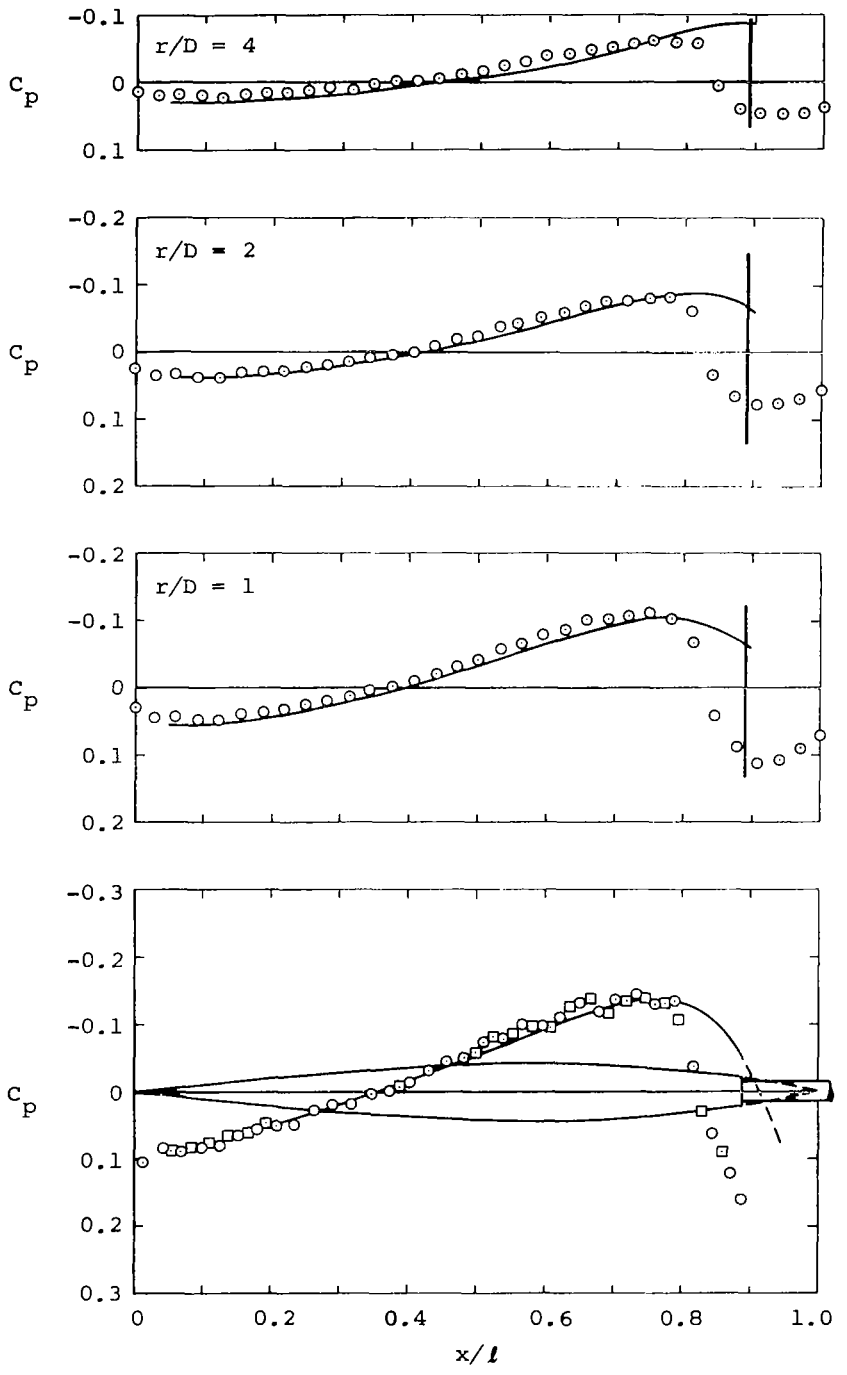
(b) $(x/\ell)_{R_{max}} = 0.4$.

Figure 4.- Continued.



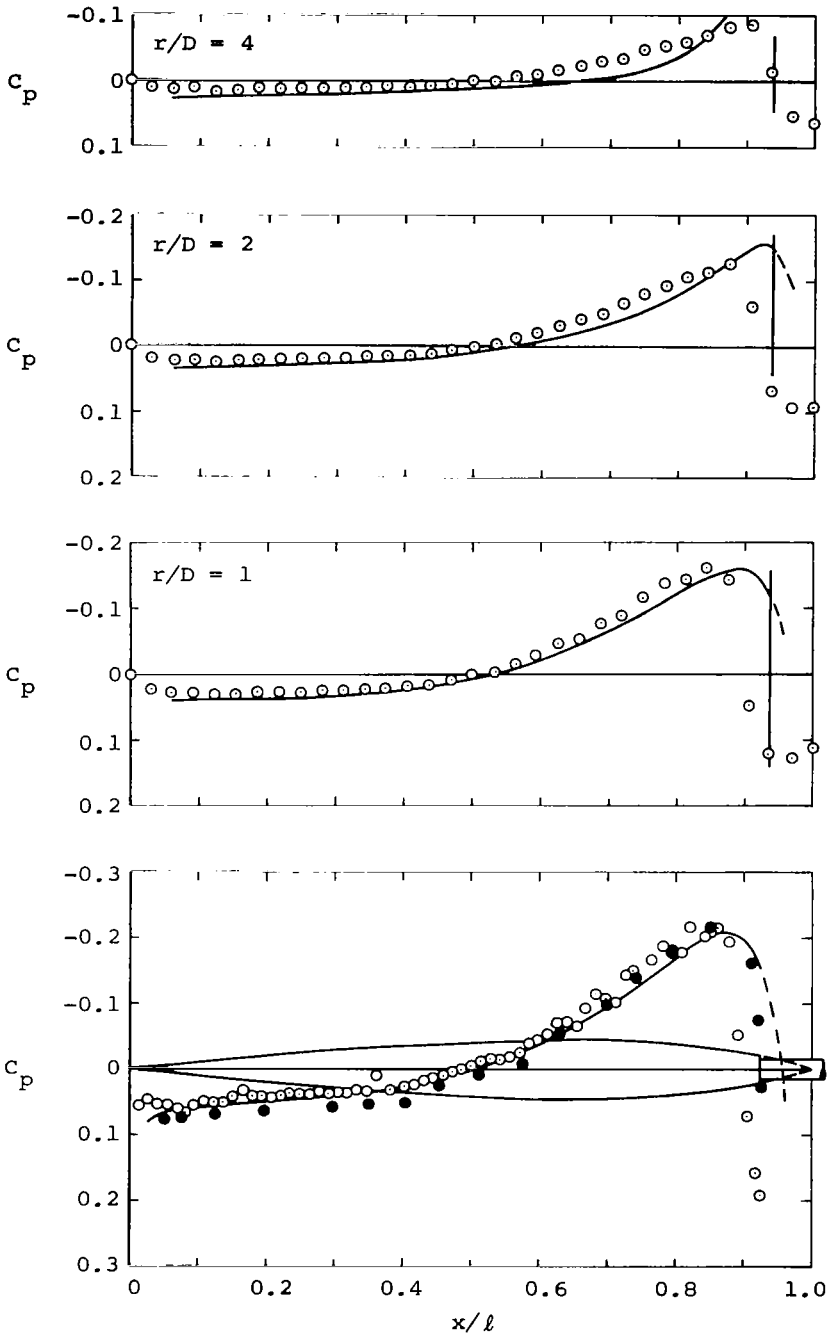
(c) $(x/\ell)_{R_{max}} = 0.5$.

Figure 4.- Continued.



(d) $(x/\ell)_{R_{max}} = 0.6$.

Figure 4.- Continued.



(e) $(x/\ell)_{R_{max}} = 0.7$.

Figure 4.- Concluded.

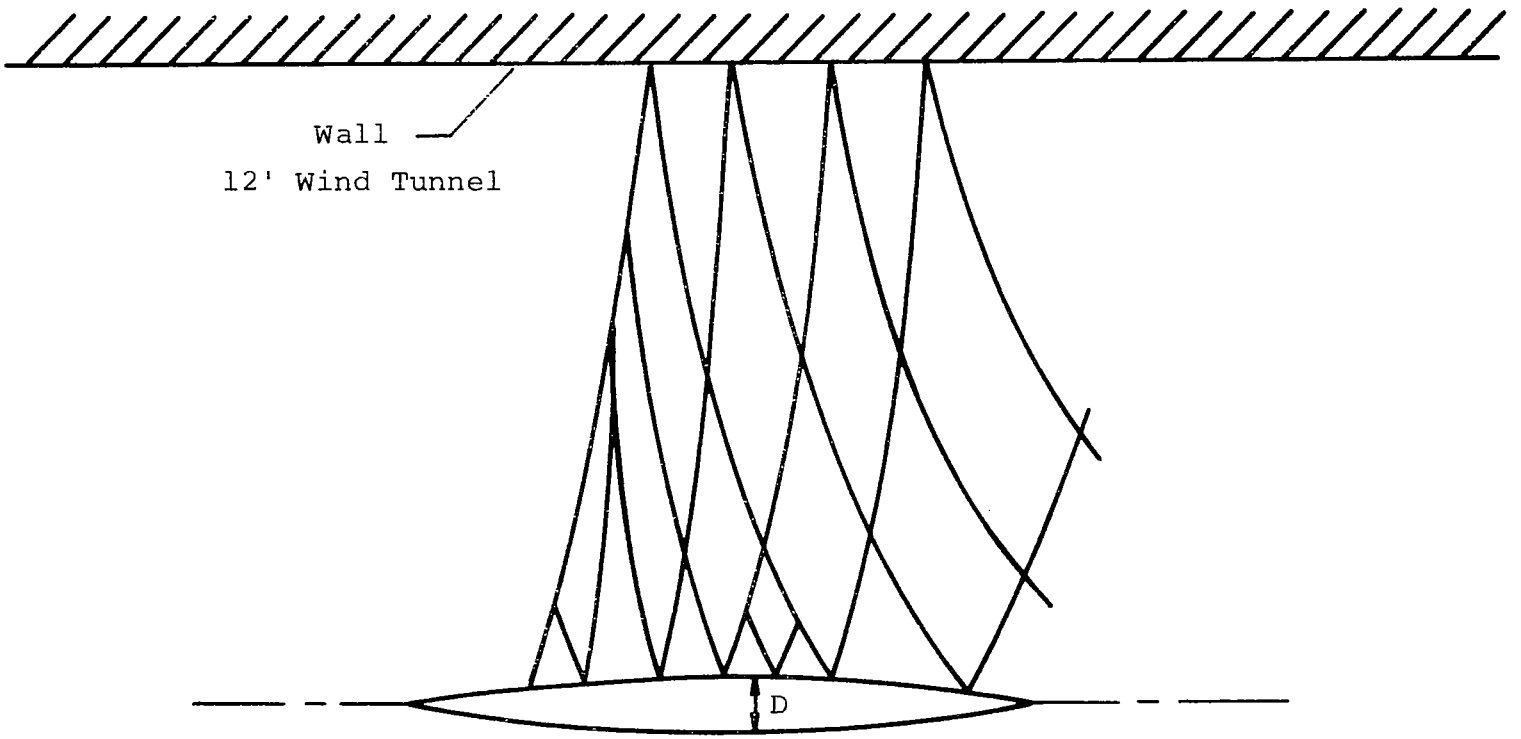
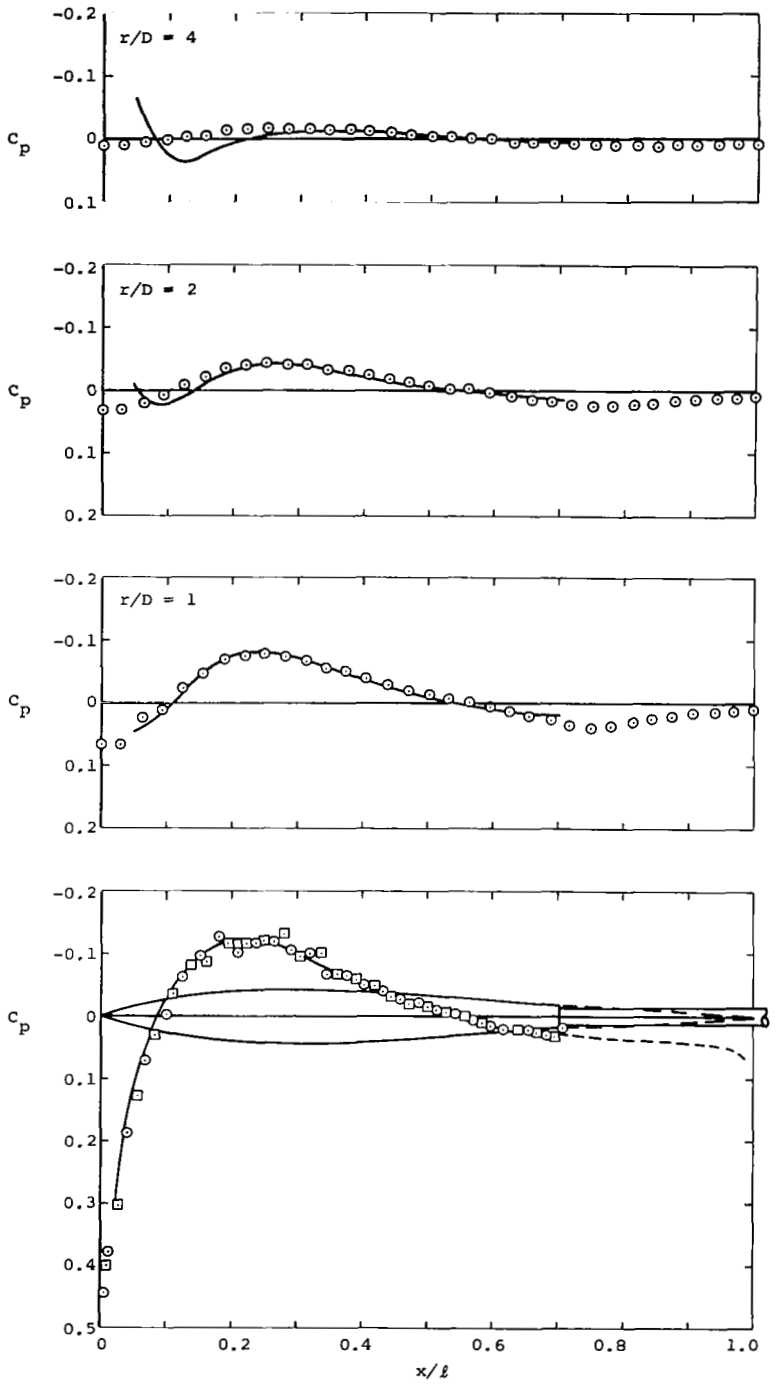
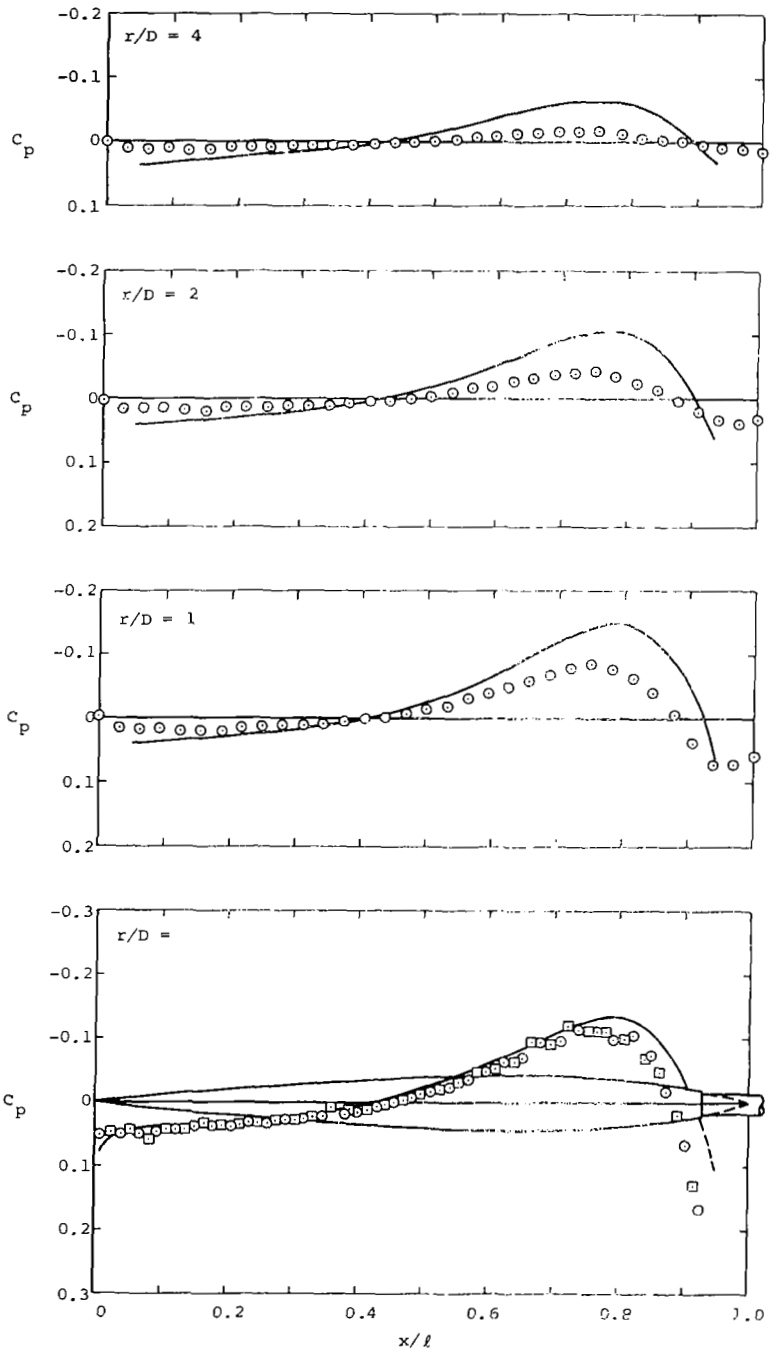


Figure 5.- Characteristic network for $M_\infty = 1$ flow past a parabolic-arc body of revolution having a diameter D = 6" and thickness ratio $D/\ell = 1/12$.



(a) $(x/\ell)_{R_{max}} = 0.3$.

Figure 6.- Theoretical and experimental surface and flow field pressure distributions for $M_\infty = 0.90$ (purely subsonic flow) over two bodies of revolution having thickness ratio $D/\ell = 1/12$ and location of the point of maximum thickness at 30 and 70 percent of the body length.



(b) $(x/\ell)_{R_{max}} = 0.7$.

Figure 6.- Concluded.

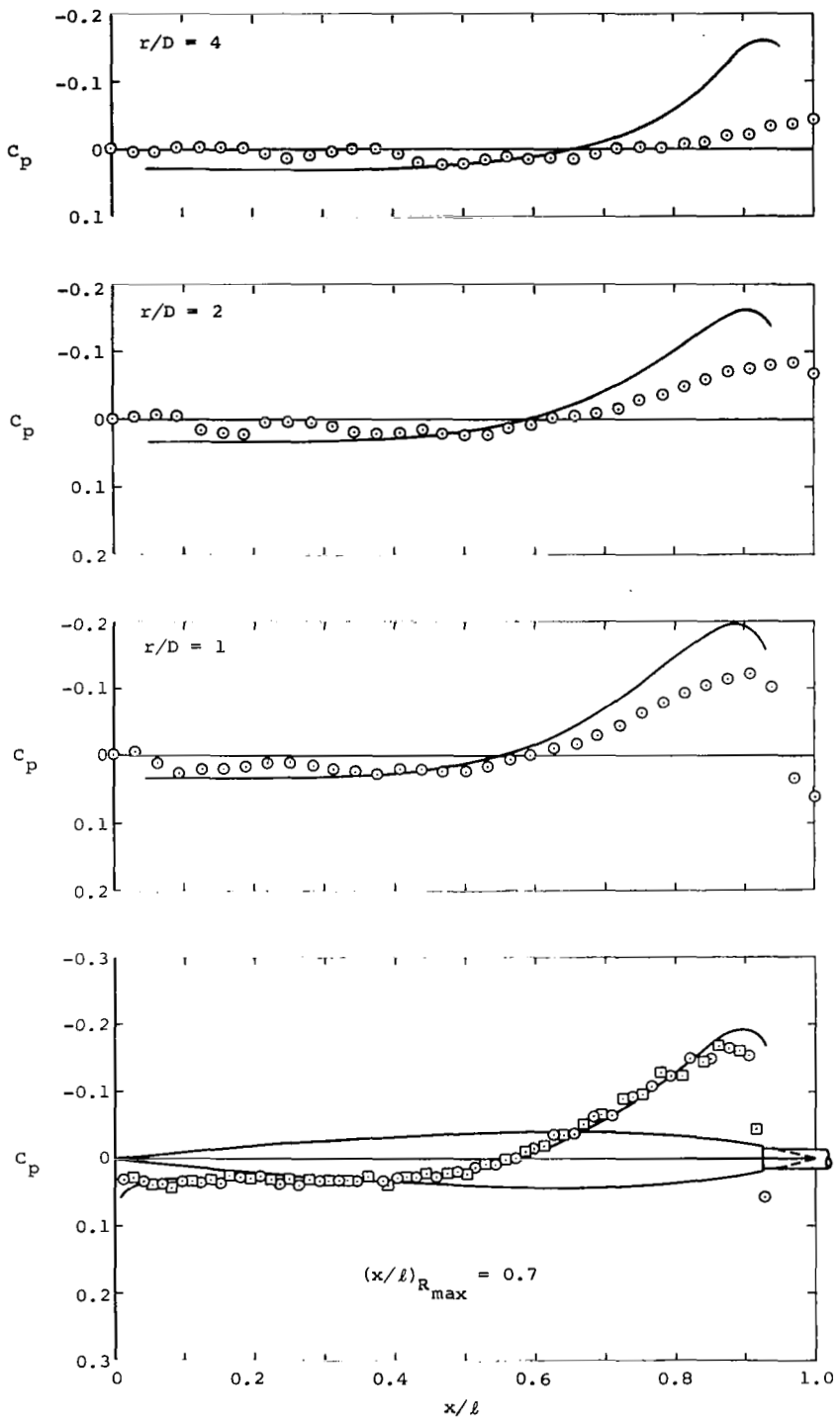


Figure 7.- Theoretical and experimental surface and flow field pressure distributions for $M_\infty = 1.20$ (purely supersonic flow) over a body of revolution having thickness ratio $D/\ell = 1/12$ and location of the point of maximum thickness at 70 percent of the body length.

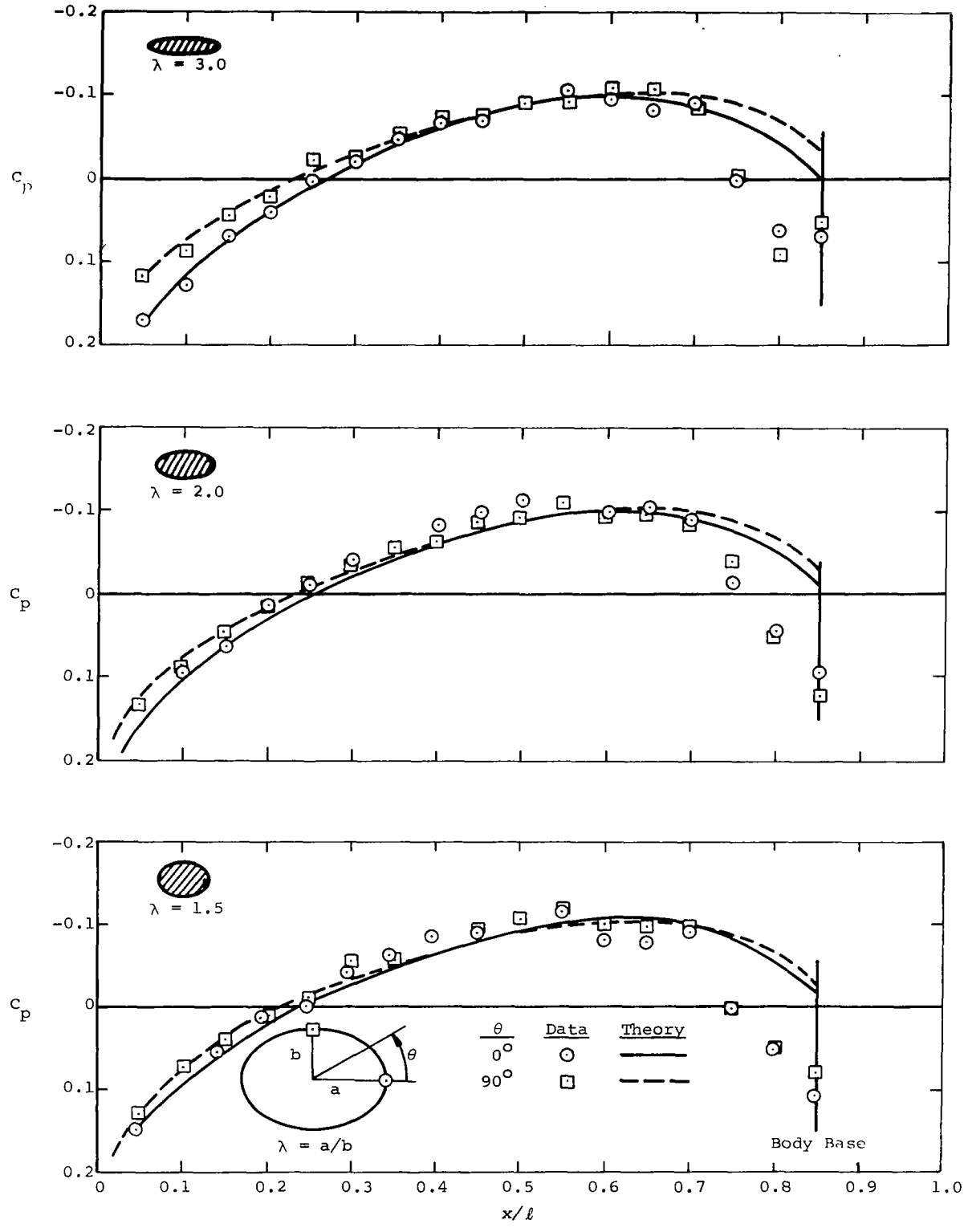


Figure 8.- Theoretical and experimental surface pressure distributions for $M_\infty = 1$ at two angular positions on three different parabolic-arc bodies having elliptical cross sections that maintain a constant value λ of the ratio of major to minor axes.

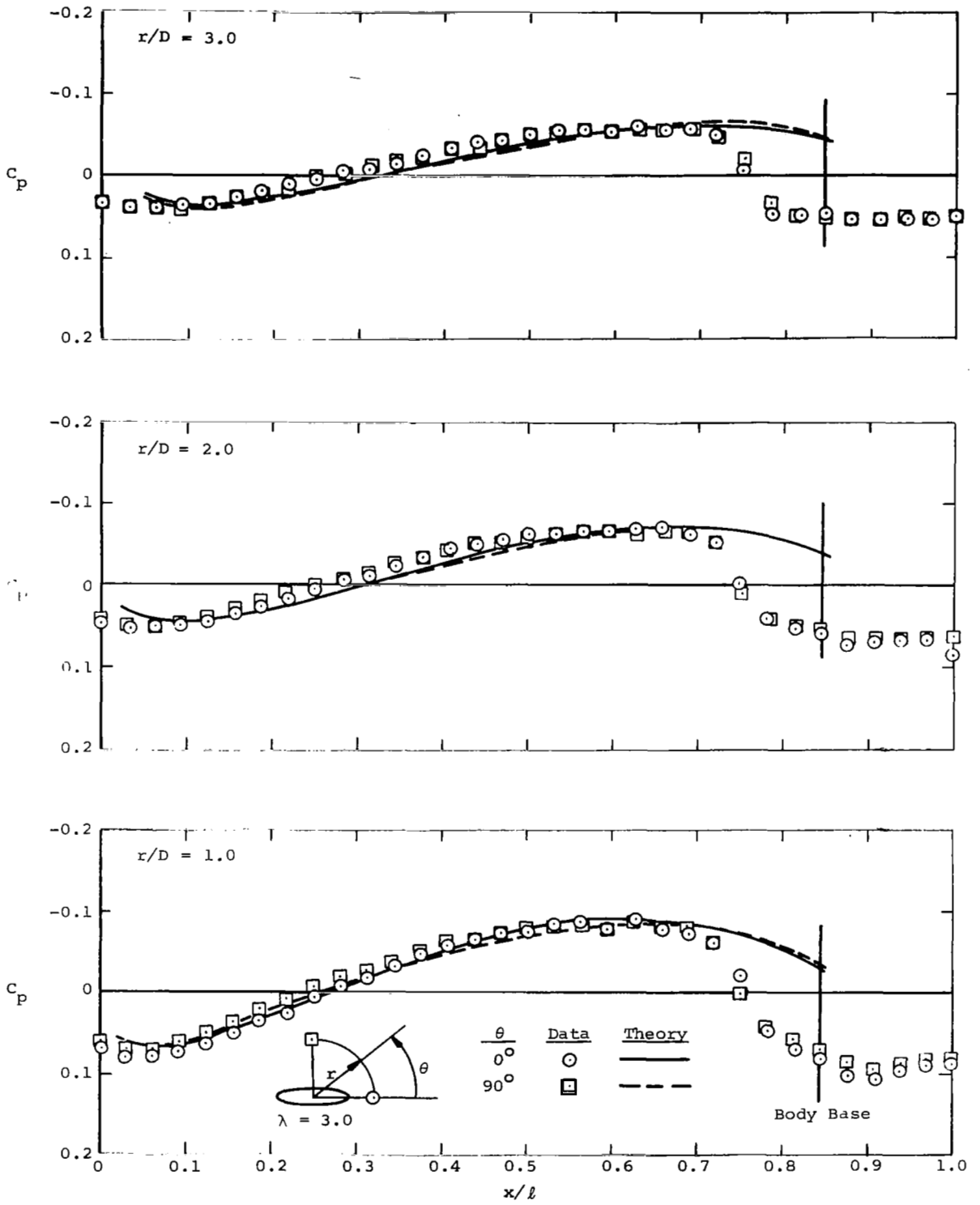


Figure 9.- Theoretical and experimental flow field pressure distributions for $M_\infty = 1$ at two angular positions and at various multiples of the maximum diameter of the equivalent body of revolution of a parabolic-arc body having an elliptic cross section with $\lambda = 3.0$.

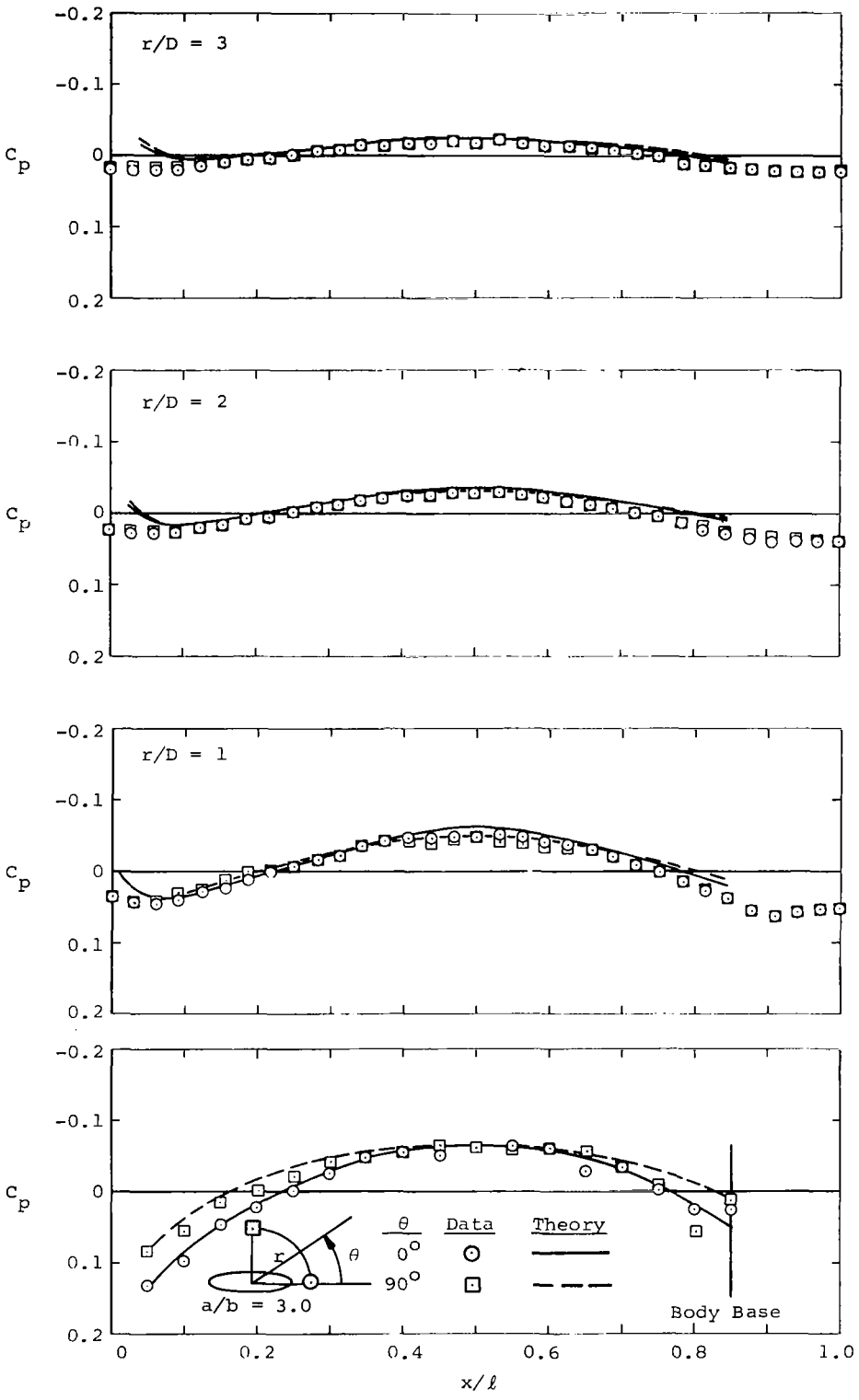


Figure 10.- Theoretical and experimental surface and flow field pressure distributions for $M_\infty = 0.90$ (purely subsonic flow) at two angular positions and three flow field locations for a parabolic-arc body with $\lambda = 3$.

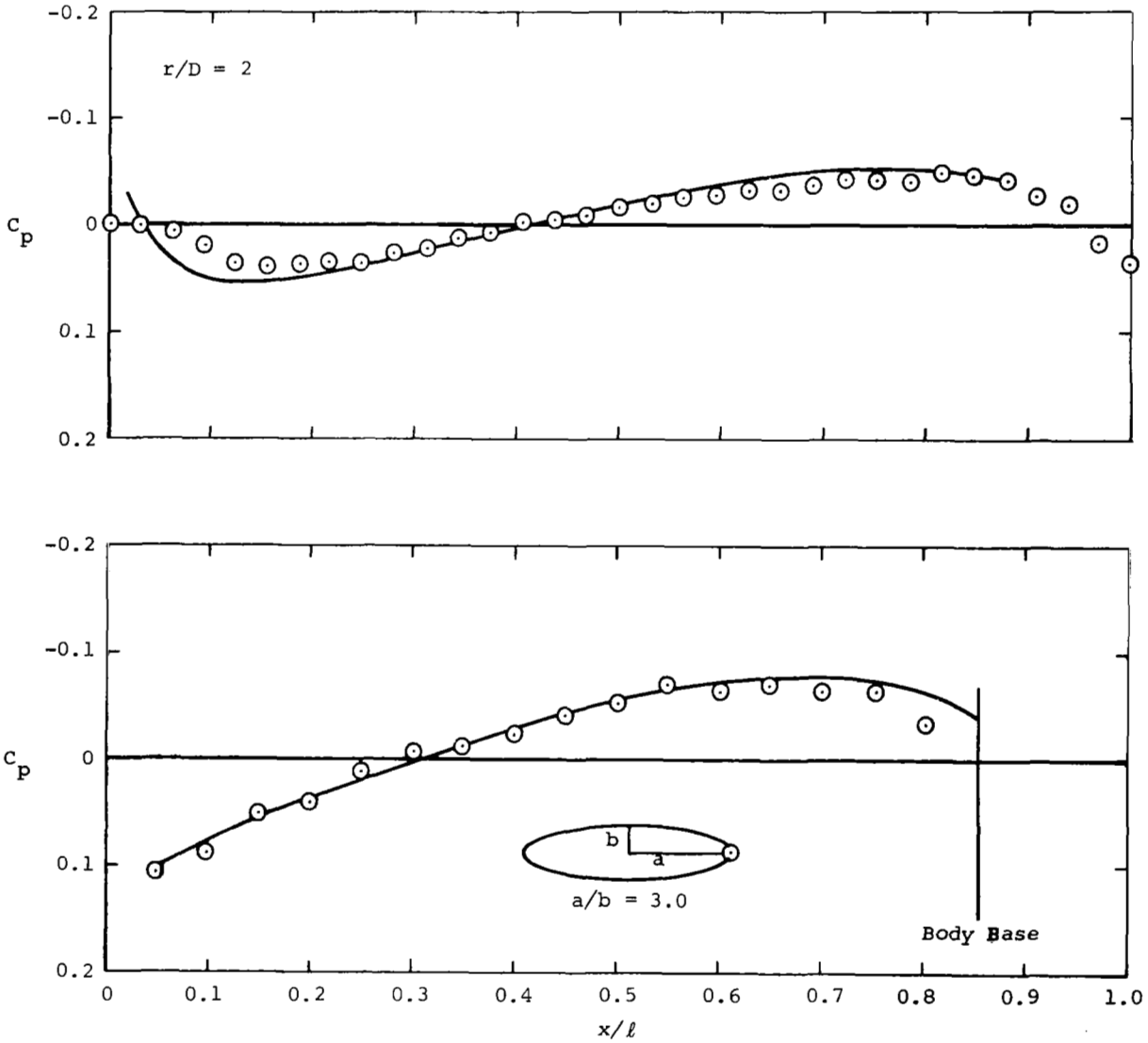


Figure 11.- Theoretical and experimental surface and flow field pressure distributions for M_∞ = 1.20 (purely supersonic flow) at one angular position and one flow field location for a parabolic-arc body with λ = 3.

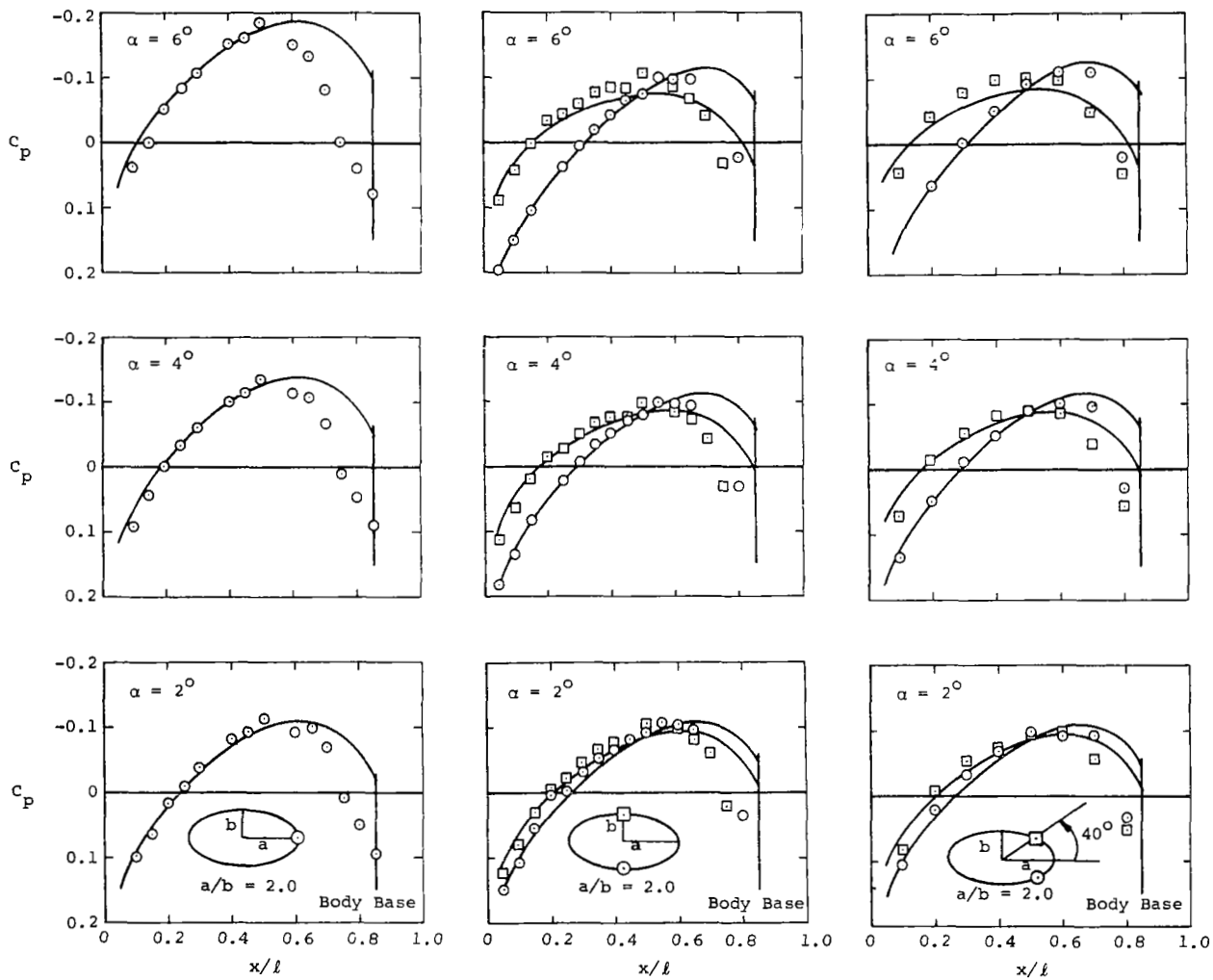


Figure 14.- Theoretical and experimental surface pressure distributions at $M_\infty = 1$ at five angular positions and three angles of attack on a parabolic-arc body having $\lambda = 2.0$ and a thickness ratio of the equivalent body of revolution of $D/\ell = 1/12$.

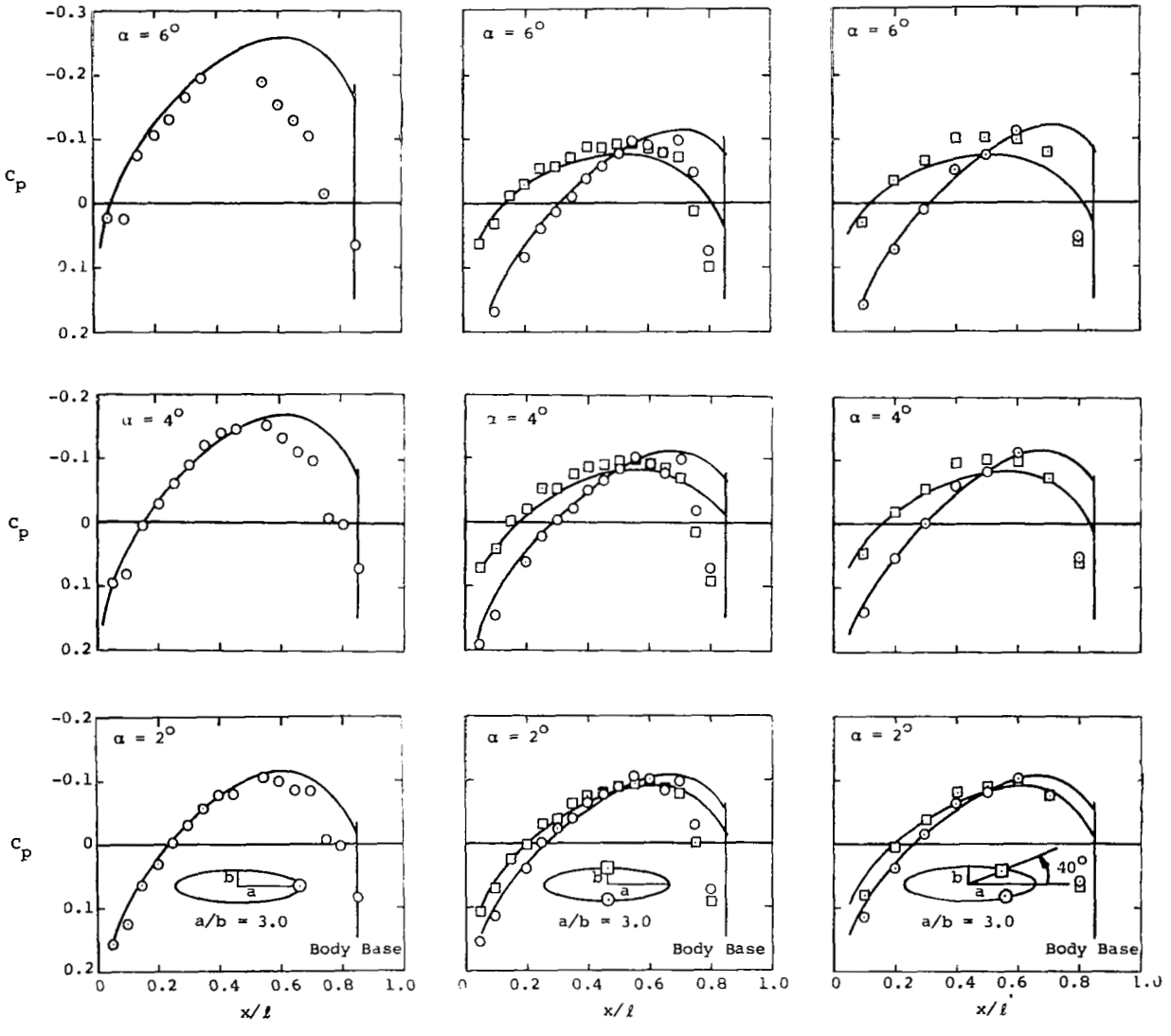


Figure 15.- Theoretical and experimental surface pressure distributions at $M_\infty = 1$ at five angular positions and three angles of attack on a parabolic-arc body having $\ell = 3.0$ and a thickness ratio of the equivalent body of revolution of $D/\ell = 1/12$.

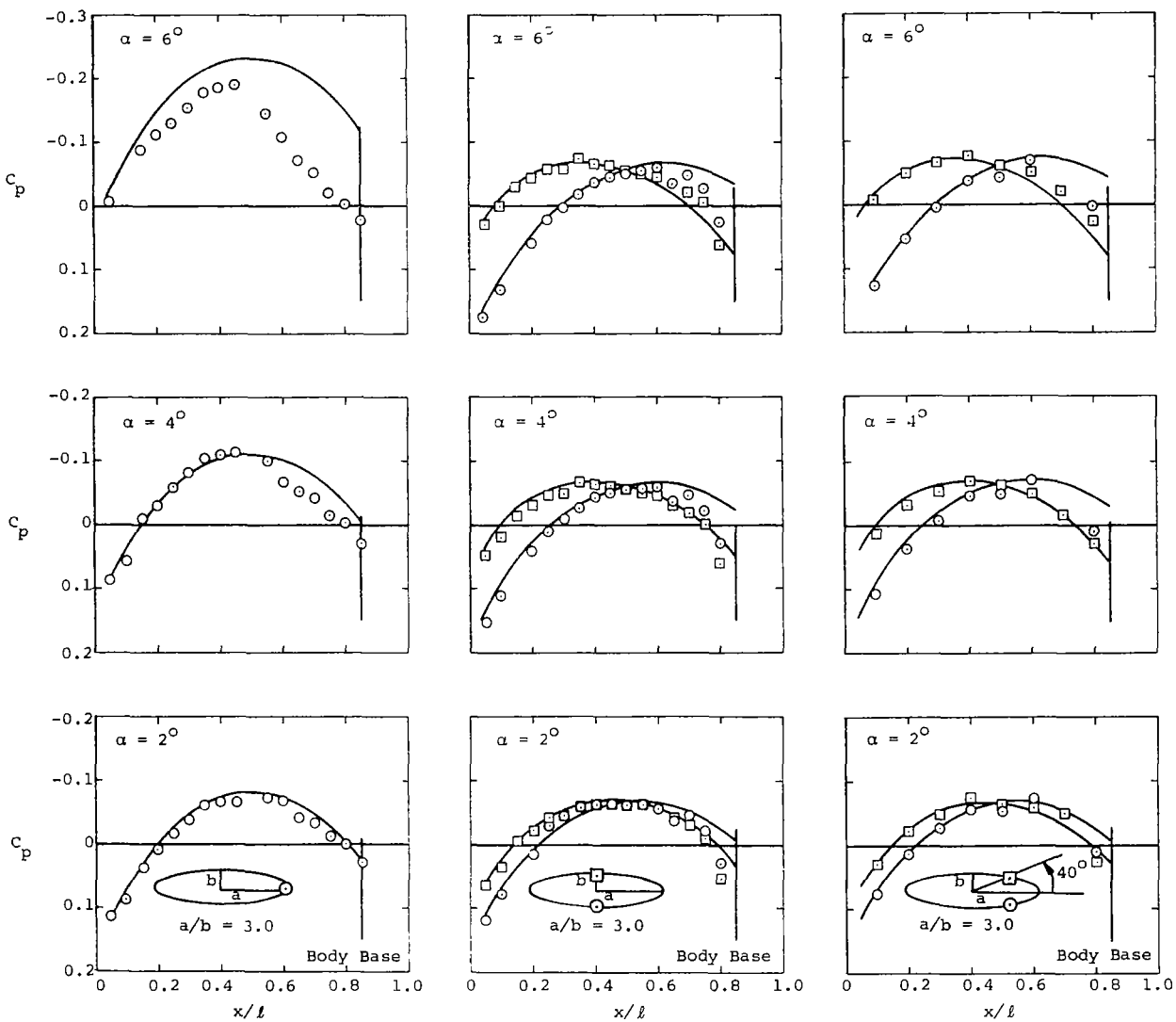


Figure 16.- Theoretical and experimental surface pressure distributions at M_∞ = 0.90 (purely subsonic flow) at five angular positions and three angles of attack on a parabolic-arc body having λ = 3.0 and a thickness ratio of the equivalent body of revolution of D/ℓ = 1/12.

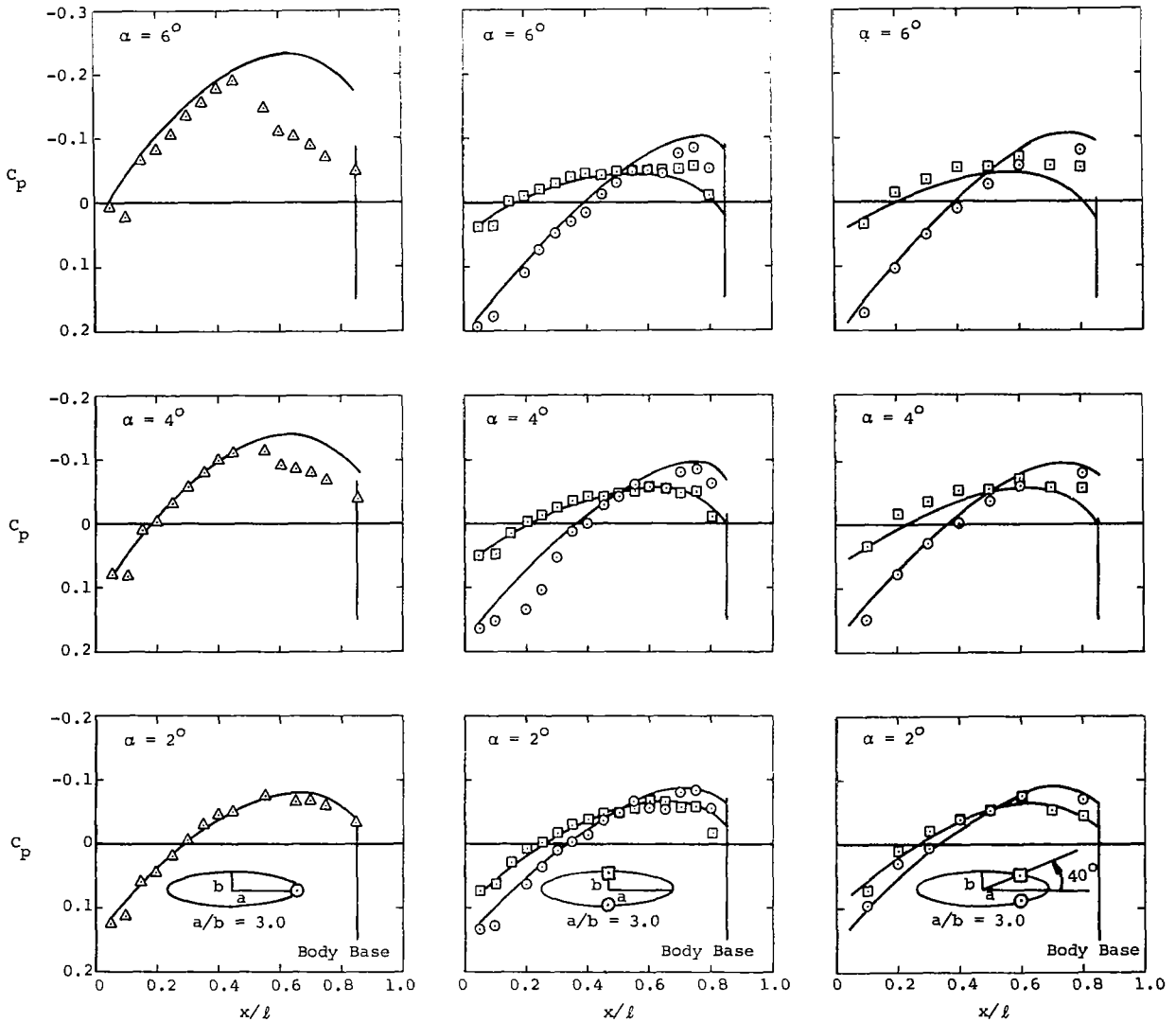


Figure 17.- Theoretical and experimental surface pressure distributions at $M_\infty = 1.20$ (purely supersonic flow) at five angular positions and three angles of attack on a parabolic-arc body having $\lambda = 3.0$ and a thickness ratio of the equivalent body of revolution of $D/\ell = 1/12$.

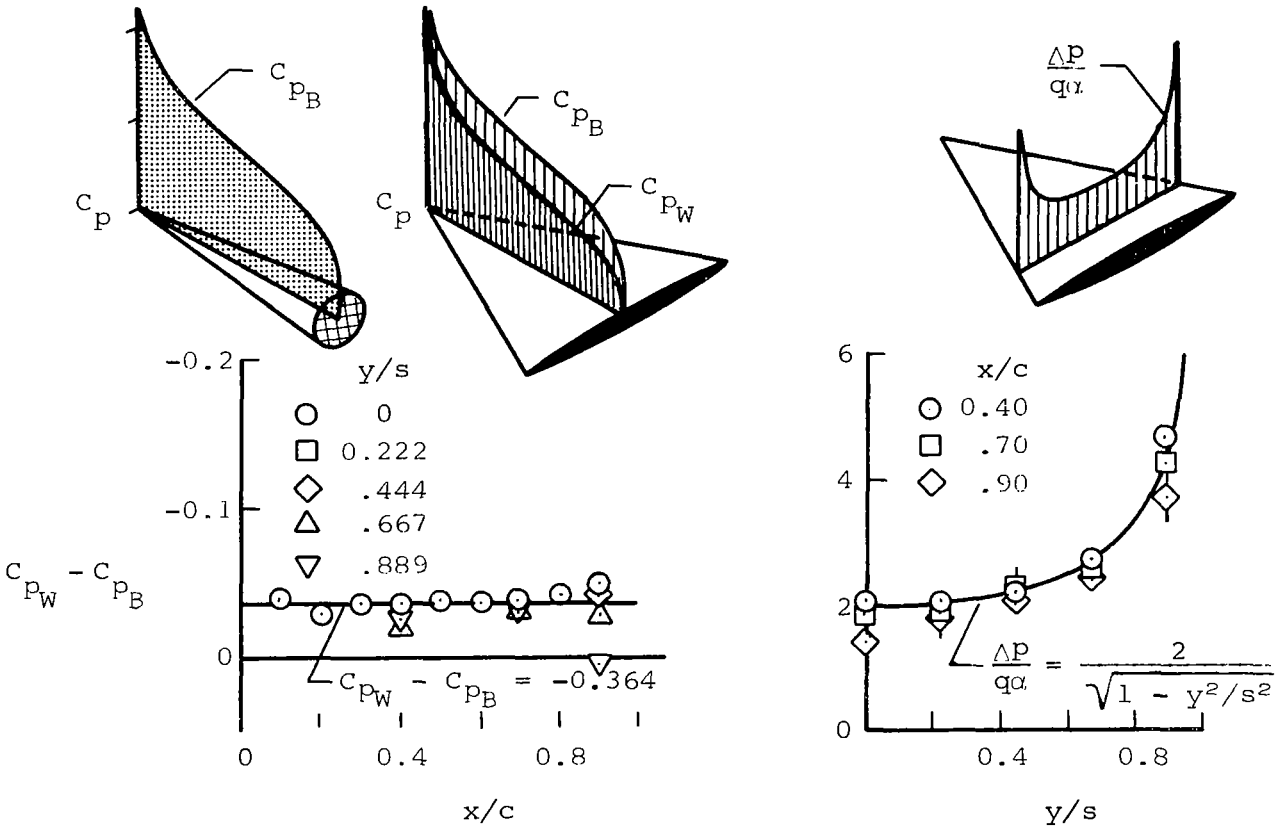


Figure 18.- Theoretical and experimental surface pressure distributions at $M_\infty = 1$ on a thin elliptic cone-cylinder similar to a triangular wing of aspect ratio 2 and thickness ratio $\tau = 0.06$.

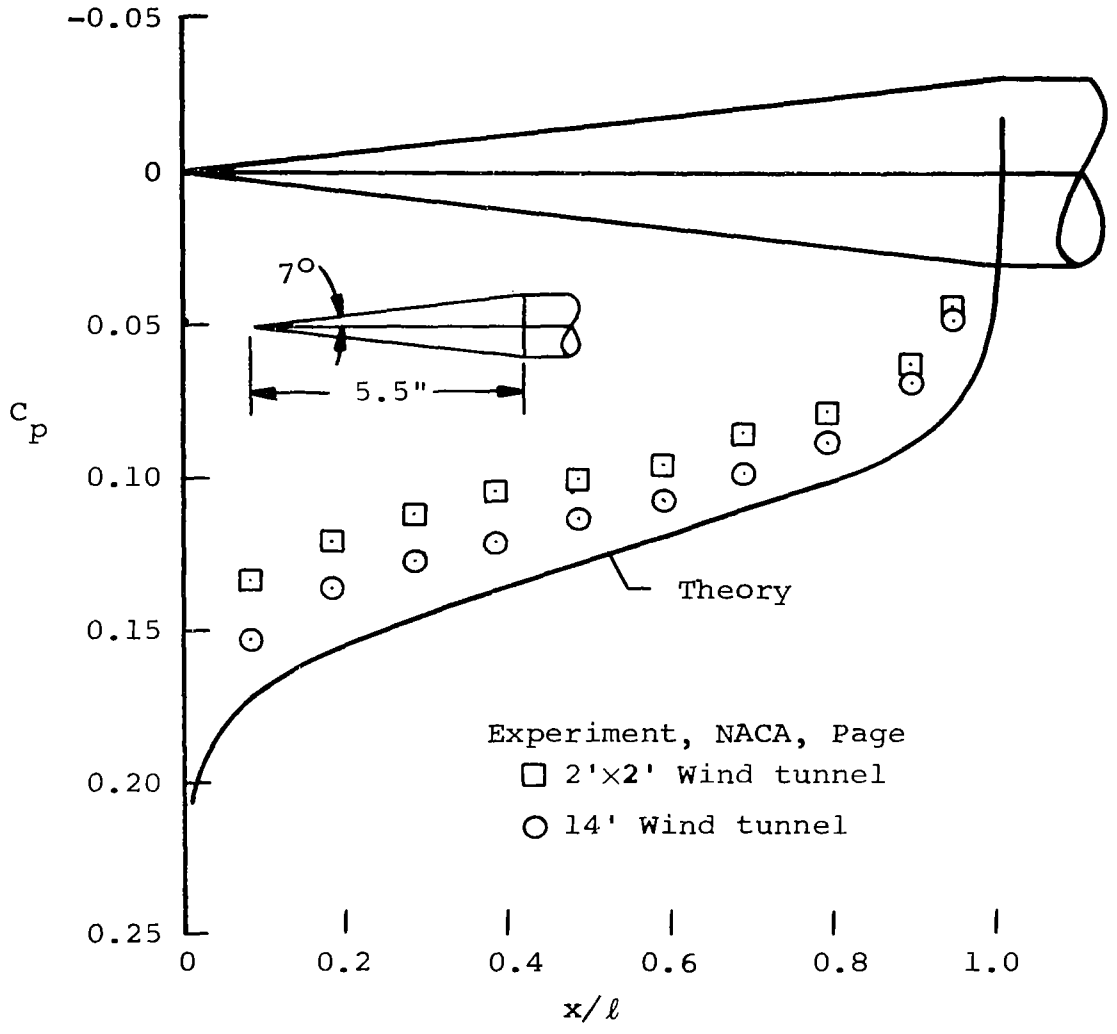


Figure 19.- Theoretical and experimental pressure distributions at $M_\infty = 1$ on a circular cone-cylinder of 7° semiapex angle and 1.35-inch diameter as measured in two transonic wind tunnels of different size and as calculated for unbounded flow.

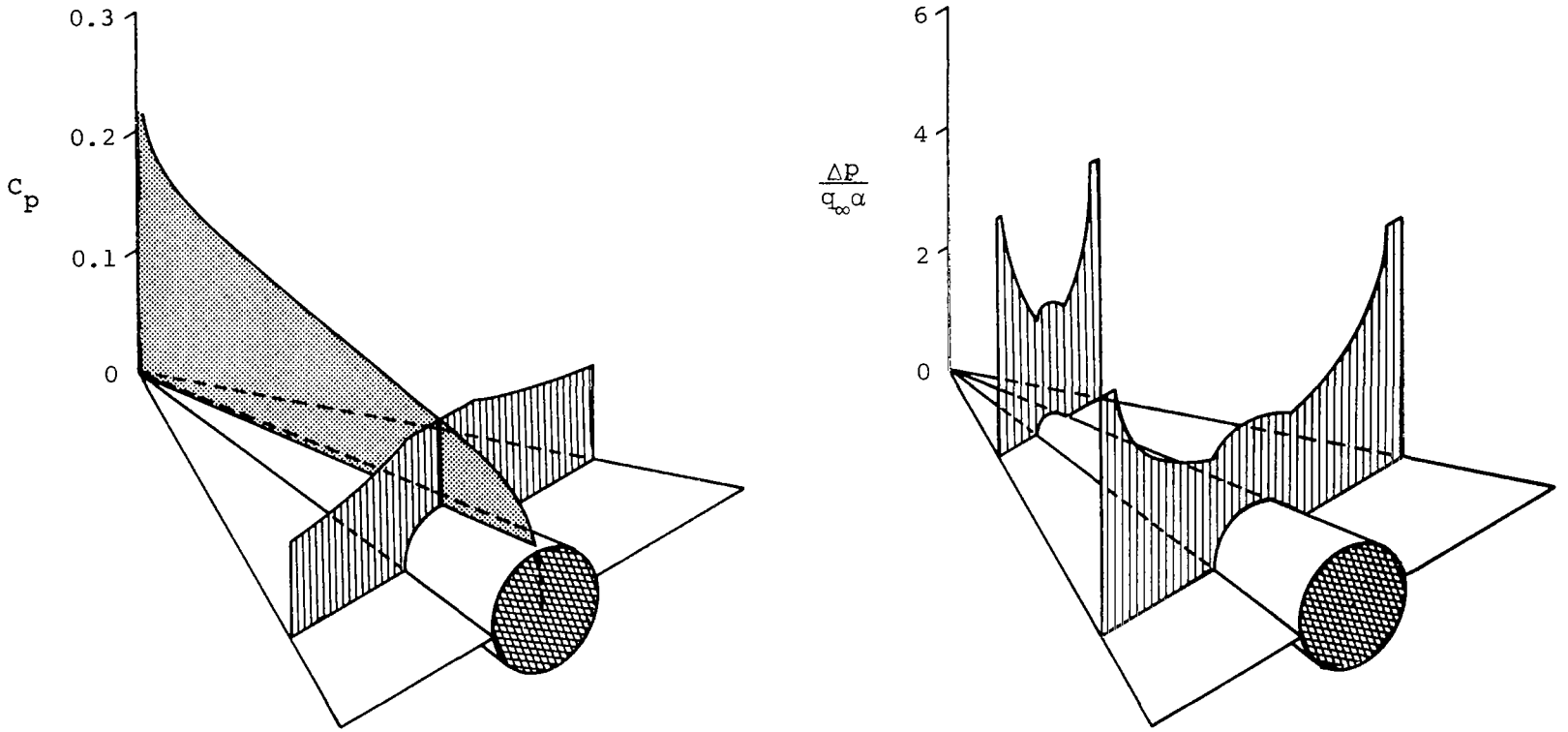


Figure 20.- Theoretical pressure distributions for $M_\infty = 1$ on a conical wing-body combination having a flat-plate triangular wing of aspect ratio 2 and a conical body of revolution having a semiapex angle of 7°.

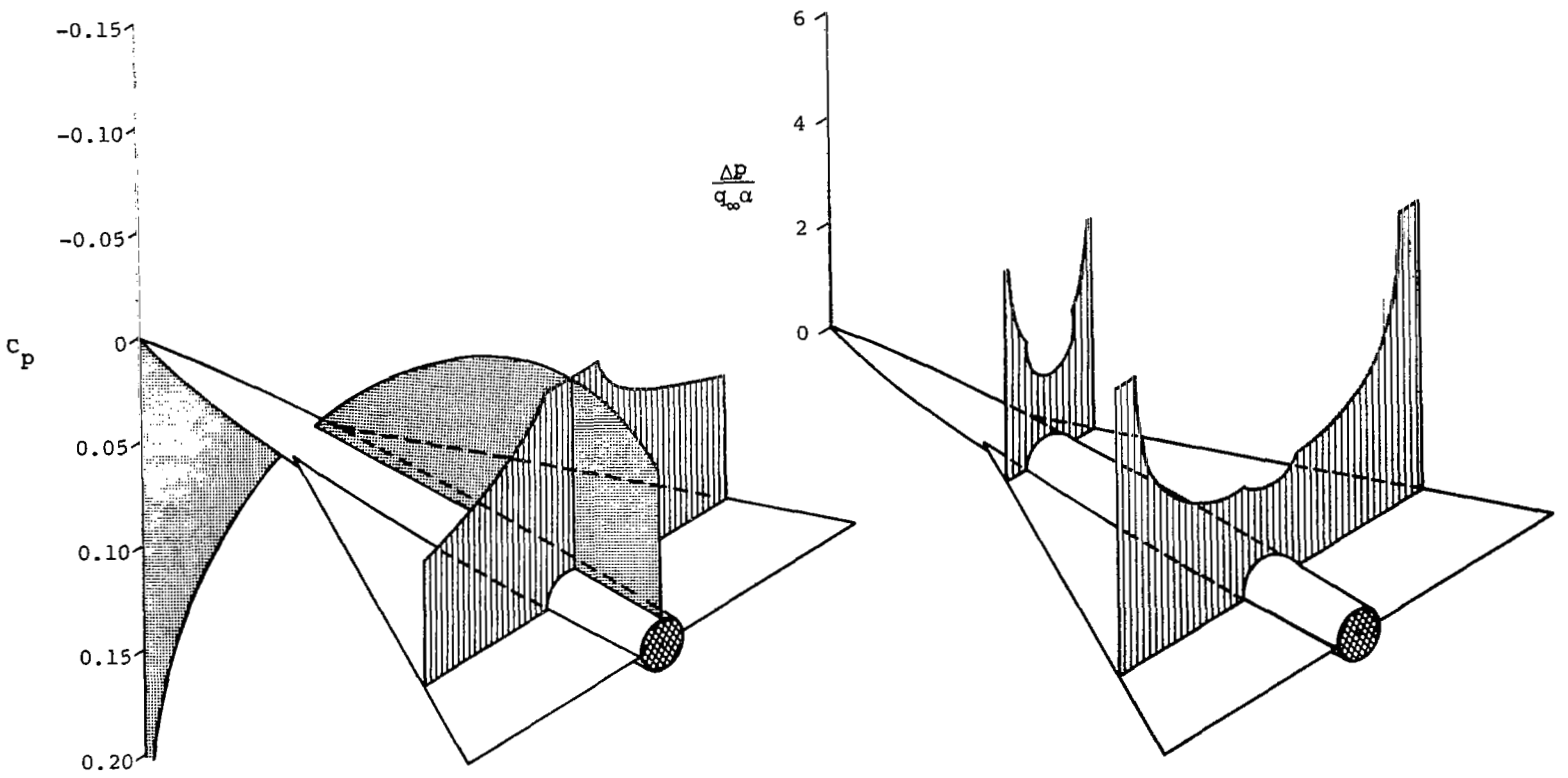


Figure 21.- Theoretical pressure distributions for $M_\infty = 1$ on a wing-body combination having a flat-plate triangular wing of aspect ratio 2 and a parabolic-arc body of revolution with thickness ratio $D/\ell = 1/12$.